VS1053b Datasheet

VS1053b - Ogg Vorbis/MP3/AAC/WMA/FLAC/ MIDI AUDIO CODEC CIRCUIT

Features

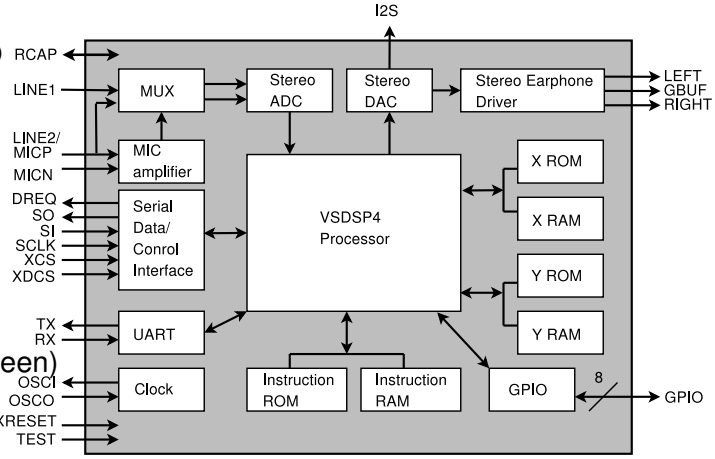
- Decodes
  Ogg Vorbis;
  MP3 = MPEG 1 & 2 audio layer III (CBR +VBR +ABR);
  MP1/MP2 = layers I & II optional;
  MPEG4 / 2 AAC-LC(+PNS),
  HE-AAC v2 (Level 3) (SBR + PS);
  WMA 4.0/4.1/7/8/9 all profiles (5-384 kbps);
  General MIDI 1 / SP-MIDI format 0 files;
  FLAC with software plugin;
  WAV (PCM + IMA ADPCM)
- Encodes Ogg Vorbis w/ software plugin
- Encodes stereo IMA ADPCM / PCM
- Streaming support for MP3 and WAV
- EarSpeaker Spatial Processing
- Bass and treble controls
- Operates with a single 12..13 MHz clock
- Can also be used with a 24..26 MHz clock
- Internal PLL clock multiplier
- Low-power operation
- High-quality on-chip stereo DAC with no phase error between channels
- Zero-cross detection for smooth volume change
- Stereo earphone driver capable of driving a 30 Ω load
- Quiet power-on and power-off
- I2S interface for external DAC
- Separate voltages for analog, digital, I/O
- On-chip RAM for user code and data
- Serial control and data interfaces
- Can be used as a slave co-processor
- SPI flash boot for special applications
- UART for debugging purposes
- New functions may be added with software and upto 8 GPIO pins
- Lead-free RoHS-compliant package (Green)

Description

VS1053b is an Ogg Vorbis/MP3/AAC/WMA/ FLAC/WAVMIDI audio decoder as well as an PCM/IMA ADPCM/Ogg Vorbis encoder on a single chip. It contains a high-performance, proprietary low-power DSP processor core VS_DSP^4, data memory, 16 KiB instruction RAM and 0.5+ KiB data RAM for user applications running simultaneously with any built-in decoder, serial control and input data interfaces, upto 8 general purpose I/O pins, an UART, as well as a high-quality variable-sample-rate stereo ADC (mic, line, line + mic or 2×line) and stereo DAC, followed by an earphone amplifier and a common voltage buffer.

VS1053b receives its input bitstream through a serial input bus, which it listens to as a system slave. The input stream is decoded and passed through a digital volume control to an 18-bit oversampling, multi-bit, sigma-delta DAC. The decoding is controlled via a serial control bus. In addition to the basic decoding, it is possible to add application specific features, like DSP effects, to the user RAM memory.

Optional factory-programmable unique chip ID provides basis for digital rights management or unit identification features.



VS1053b音频编解码芯片 - Ogg Vorbis/MP3/AAC/WMA/FLAC/MIDI音频编解码电路

特性

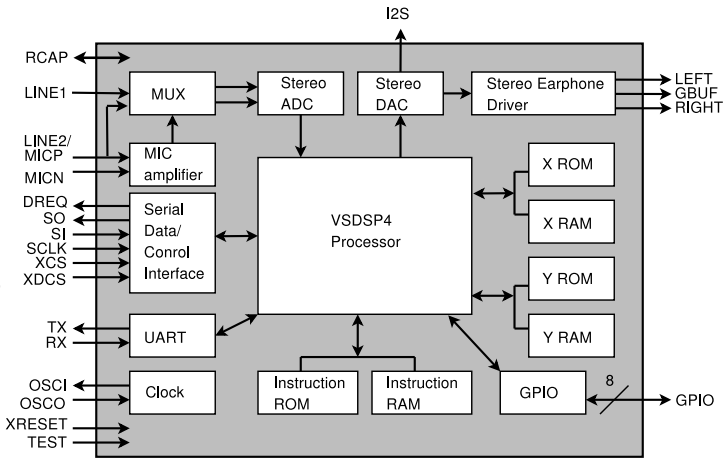
- 解码支持
  Ogg Vorbis;
  MP3 = MPEG 1 & 2 音频层III (固定比特率+可变比特率+平均比特率);
  MP1/MP2 = 可选支持层I & II;
  MPEG4 / 2 AAC-LC(+PNS),
  HE-AAC v2（第3级）（SBR + PS）;
  WMA 4.0/4.1/7/8/9全规格支持 (5-384 kbps);
  通用MIDI 1 / SP-MIDI格式0文件;
  需软件插件支持FLAC解码;
  WAV音频格式（PCM + IMA ADPCM）
- 通过软件插件编码Ogg Vorbis
- 编码立体声IMA ADPCM / PCM
- 支持MP3和WAV流媒体传输
- EarSpeaker空间音频处理技术
- 低音与高音控制
- 工作时钟频率为12..13 MHz单一时钟
- 也可使用24..26 MHz时钟
- 内部PLL时钟倍频器
- 低功耗运行
- 高质量片上立体声DAC，通道间无相位差
- 零交叉检测实现平滑音量调节
- 立体声耳机驱动器，可驱动30Ω负载
- 静音上电与断电
- 用于外接DAC的I2S接口
- 模拟、数字和I/O独立供电
- 用户代码与数据存储的片上RAM
- 串行控制与数据接口
- 可用作从协处理器
- 特殊应用的SPI闪存启动
- 调试用UART接口
- 可通过软件和最多8个GPIO引脚扩展新功能
- 符合无铅RoHS标准的封装（绿色）

描述

VS1053b音频编解码芯片是一款单芯片解决方案，集成了Ogg Vorbis/MP3/AAC/WMA/FLAC/WAV/MIDI音频解码功能，以及PCM/IMA ADPCM/Ogg Vorbis编码功能。该芯片包含高性能、专有的低功耗DSP处理器核心VS_DSP4，数据存储器，16 KiB指令RAM和0.5 + KiB数据RAM，用于用户应用程序与任何内置解码器同步运行，串行控制和输入数据接口，最多8个通用I/O引脚，一个UART，以及高质量可变采样率立体声ADC（麦克风、线路、线路+麦克或2×线路输入）和立体声DAC，后接耳机放大器和公共电压缓冲器。

VS1053b通过串行输入总线接收输入比特流，作为系统从设备进行监听。输入流经解码后通过数字音量控制传递至18位过采样、多位Σ-Δ DAC。解码过程通过串行控制总线进行控制。除基本解码功能外，还可向用户RAM内存添加特定应用功能，如DSP音效处理。

可选的出厂可编程唯一芯片ID，为数字版权管理或设备识别功能提供基础。



Contents

内容

List of Figures

图目录

1 Licenses

MPEG Layer-3 audio decoding technology licensed from Fraunhofer IIS and Thomson.

Note: If you enable Layer I and Layer II decoding, you are liable for any patent issues that may arise from using these formats. Joint licensing of MPEG 1.0 / 2.0 Layer III does not cover all patents pertaining to layers I and II.

VS1053b contains WMA decoding technology from Microsoft.
This product is protected by certain intellectual property rights of Microsoft and cannot be used or further distributed without a license from Microsoft.

VS1053b contains AAC technology (ISO/IEC 13818-7 and ISO/IEC 14496-3) which cannot be used without a proper license from Via Licensing Corporation or individual patent holders.

VS1053b contains spectral band replication (SBR) and parametric stereo (PS) technologies developed by Coding Technologies. Licensing of SBR is handled within MPEG4 through Via Licensing Corporation. Licensing of PS is handled with Coding Technologies.
See *http://www.codingtechnologies.com/licensing/aacplus.htm* for more information.

To the best of our knowledge, if the end product does not play a specific format that otherwise would require a customer license: MPEG 1.0/2.0 layers I and II, WMA, or AAC, the respective license should not be required. Decoding of MPEG layers I and II are disabled by default, and WMA and AAC format exclusion can be easily performed based on the contents of the SCI_HDAT1 register. Also PS and SBR decoding can be separately disabled.

2 Disclaimer

All properties and figures are subject to change.

3 Definitions

B Byte, 8 bits.

b Bit.

Ki "Kibi" = 2^{10} = 1024 (IEC 60027-2).

Mi "Mebi" = 2^{20} = 1048576 (IEC 60027-2).

VS_DSP VLSI Solution's DSP core.

W Word. In VS_DSP, instruction words are 32-bit and data words are 16-bit wide.

1 许可协议

MPEG Layer-3音频解码技术授权自Fraunhofer IIS与Thomson公司

注意：若启用Layer I和Layer II解码功能，使用者须自行承担因使用这些格式可能产生的专利法律责任MPEG 1.0/2.0 Layer III的联合许可不涵盖涉及Layer I和II的所有专利。

VS1053b包含来自微软的WMA解码技术。
本产品受微软特定知识产权保护，未经微软授权不得使用或进一步分发。

VS1053b包含AAC技术（ISO/IEC 13818-7与ISO/IEC 14496-3），未经Via Licensing公司或个别专利持有者的适当许可不得使用。

VS1053b包含由Coding Technologies开发的频带复制(SBR)和参数立体声(PS)技术。SBR许可通过Via Licensing公司在MPEG4框架内处理，PS许可需与Coding Technologies协商。

更多信息请参见*http://www.codingtechnologies.com/licensing/aacplus.htm*

据我们所知，若终端产品不播放可能需客户单独授权的特定格式：MPEG 1.0/2.0 Layer I/II、WMA或AAC，则无需取得相应许可。默认情况下禁用MPEG第I层和第II层解码，且可根据SCI_HDAT1寄存器的内容轻松执行WMA和AAC格式排除。同时也可单独禁用PS与SBR解码功能。

2 免责声明

所有属性及数值均可能变更。

3 定义说明

B 字节，8位。

b 比特。

Ki “千位二进制” = 2^{10}= 1024（符合IEC 60027-2标准）。

Mi “兆位二进制” = 2^{20}= 1048576（符合IEC 60027-2标准）。

VS_DSP VLSI Solution公司数字信号处理器核心。

W 字。在VS_DSP中，指令字为32位宽，数据字为16位宽。

4 Characteristics & Specifications

4.1 Absolute Maximum Ratings

Parameter	Symbol	Min	Max	Unit
Analog Positive Supply	AVDD	-0.3	3.6	V
Digital Positive Supply	CVDD	-0.3	1.85	V
I/O Positive Supply	IOVDD	-0.3	3.6	V
Current at Any Non-Power Pin[1]			$\pm$50	mA
Voltage at Any Digital Input		-0.3	IOVDD+0.3[2]	V
Operating Temperature		-30	+85	°C
Storage Temperature		-65	+150	°C

[1] Higher current can cause latch-up.
[2] Must not exceed 3.6 V

4.2 Recommended Operating Conditions

Parameter	Symbol	Min	Typ	Max	Unit
Ambient Operating Temperature		-30		+85	°C
Analog and Digital Ground [1]	AGND DGND		0.0		V
Positive Analog, REF=1.23V	AVDD	2.5	2.8	3.6	V
Positive Analog, REF=1.65V [2]	AVDD	3.3	3.3	3.6	V
Positive Digital	CVDD	1.7	1.8	1.85	V
I/O Voltage	IOVDD	1.8	2.8	3.6	V
Input Clock Frequency [3]	XTALI	12	12.288	13	MHz
Internal Clock Frequency	CLKI	12	36.864	55.3	MHz
Internal Clock Multiplier [4]		$1.0\times$	$3.0\times$	$4.5\times$	
Master Clock Duty Cycle		40	50	60	%

[1] Must be connected together as close the device as possible for latch-up immunity.
[2] Reference voltage can be internally selected between 1.23V and 1.65V, see section 9.6.2.
[3] The maximum sample rate that can be played with correct speed is XTALI/256 (or XTALI/512 if SM_CLK_RANGE is set). Thus, XTALI must be at least 12.288 MHz (24.576 MHz) to be able to play 48 kHz at correct speed.
[4] Reset value is $1.0\times$. Recommended SC_MULT=$3.5\times$, SC_ADD=$1.0\times$ (SCI_CLOCKF=0x8800). Do not exceed maximum specification for CLKI.

4 特性与规格说明

4.1 绝对最大额定值

参数	符号	最小值	最大值	单位
模拟正电源供电	特定术语：AVDD	-0.3	3.6	V
数字正电源供电	特定术语：CVDD	-0.3	1.85	V
输入/输出正电源供电	IOVDD	-0.3	3.6	V
任意非电源引脚电流[1]			±50	mA
任意数字输入电压		-0.3	IOVDD+0.3[2]	V
工作温度		-30	+85	°C
存储温度		-65	+150	°C

[1] 过高电流可能引发闩锁效应
[2] 不得超过3.6V

4.2 推荐工作条件

参数	符号	最小值	典型值	最大值	单位
工作环境温度		-30		+85	°C
模拟与数字地 [1]	AGND DGND		0.0		V
模拟正极供电，参考电压=1.23V	特定术语：AVDD	2.5	2.8	3.6	V
模拟正极供电，参考电压=1.65V [2]	特定术语：AVDD	3.3	3.3	3.6	V
数字正极供电	特定术语：CVDD	1.7	1.8	1.85	V
I/O电压	IOVDD	1.8	2.8	3.6	V
输入时钟频率 [3]	XTALI	12	12.288	13	兆赫兹
内部时钟频率	特定术语：CLKI	12	36.864	55.3	兆赫兹
内部时钟倍频器 [4]		1.0×	3.0×	4.5×	
主时钟占空比		40	50	60	%

[1] 为增强抗闩锁能力，必须尽可能靠近器件将两地短接
[2] 参考电压可在1.23V与1.65V之间内部选择，详见章节9.6.2
[3] 能以正确速度播放的最高采样率为 XTALI/256（若设置 SM_CLK_RANGE 则为 XTALI/512）。 因此，要能以正确速度播放 48 千赫兹音频，晶振频率 XTALI 至少需为 12.288 兆赫兹（24.576 兆赫兹）。
[4] 复位值为 1.0×倍。推荐配置：SC_MULT=3.5×倍，SC_ADD=1.0×倍（SCI_CLOCKF=0x8800）。切勿超过 CLKI 的最大规格限值。

4.3 Analog Characteristics

Unless otherwise noted: AVDD=3.3V, CVDD=1.8V, IOVDD=2.8V, REF=1.65V, TA=-30..+85°C, XTALI=12..13MHz, Internal Clock Multiplier $3.5\times$. DAC tested with 1307.894 Hz full-scale output sinewave, measurement bandwidth 20..20000 Hz, analog output load: LEFT to GBUF 30 Ω, RIGHT to GBUF 30 Ω. Microphone test amplitude 48 mVpp, f_s=1 kHz, Line input test amplitude 1.26 V, f_s=1 kHz.

Parameter	Symbol	Min	Typ	Max	Unit
DAC Resolution			18		bits
Total Harmonic Distortion	THD			0.07	%
Third Harmonic Distortion				0.02	%
Dynamic Range (DAC unmuted, A-weighted)	IDR		100		dB
S/N Ratio (full scale signal)	SNR		94		dB
Interchannel Isolation (Cross Talk), 600Ω + GBUF			80		dB
Interchannel Isolation (Cross Talk), 30Ω + GBUF			53		dB
Interchannel Gain Mismatch		-0.5		0.5	dB
Frequency Response		-0.1		0.1	dB
Full Scale Output Voltage (Peak-to-peak)		1.64	1.85[1]	2.06	Vpp
Deviation from Linear Phase				5	°
Analog Output Load Resistance	AOLR	16	30[2]		Ω
Analog Output Load Capacitance				100	pF
Microphone input amplifier gain	MICG		26		dB
Microphone input amplitude			48	140[3]	mVpp AC
Microphone Total Harmonic Distortion	MTHD		0.03	0.07	%
Microphone S/N Ratio	MSNR	60	70		dB
Microphone input impedances, per pin			45		kΩ
Line input amplitude			2500	2800[3]	mVpp AC
Line input Total Harmonic Distortion	LTHD		0.005	0.014	%
Line input S/N Ratio	LSNR	85	90		dB
Line input impedance			80		kΩ

[1] 3.0 volts can be achieved with +-to-+ wiring for mono difference sound.
[2] AOLR may be much lower, but below *Typical* distortion performance may be compromised.
[3] Above typical amplitude the Harmonic Distortion increases.

4.3 模拟特性

除非另有说明： AVDD=3.3伏， CVDD=1.8伏， IOVDD=2.8伏， REF=1.65伏， TA=-30～+85°C，XTALI=12～13兆赫兹，内部时钟倍频器3.5×倍。DAC测试采用1307.894赫兹满量程输出正弦波，测量带宽20..20000赫兹，模拟输出负载：左声道至GBUF 30欧姆，右声道至GBUF 30欧姆。 麦克风测试幅值48毫伏峰峰值，频率=1千赫兹；线路输入测试幅值1.26伏，频率=1千赫兹。

参数	符号	最小值	典型值	最大值	单位
DAC分辨率			18		位
总谐波失真	THD			0.07	%
三次谐波失真				0.02	%
动态范围（DAC非静音，A计权）	IDR		100		分贝
信噪比（满量程信号）	SNR		94		分贝
通道间隔离度（串扰），600欧姆+ GBUF			80		分贝
通道间隔离度（串扰），30欧姆+ GBUF			53		分贝
通道间增益失配		-0.5		0.5	分贝
频率响应		-0.1		0.1	分贝
满量程输出电压（峰峰值）		1.64	1.85[1]	2.06	伏峰峰值
线性相位偏差				5	°
模拟输出负载电阻	AOLR	16	30[2]		Ω
模拟输出负载电容				100	pF
麦克风输入放大器增益	MICG		26		分贝
麦克风输入幅度			48	140[3]	毫伏峰峰值交流
麦克风总谐波失真	MTHD		0.03	0.07	%
麦克风信噪比	MSNR	60	70		分贝
单引脚麦克风输入阻抗			45		千Ω
线路输入幅度			2500	2800[3]	毫伏峰峰值交流
线路输入总谐波失真	LTHD		0.005	0.014	%
线路输入信噪比	LSNR	85	90		分贝
线路输入阻抗			80		千Ω

[1] 采用单声道差分音频的正负接线方式，可实现3.0伏特输出电压。
[2] AOLR可能显著降低，但低于典型值时失真性能可能劣化。
[3] 超过典型振幅时谐波失真增加。

4.4 Power Consumption

Tested with an Ogg Vorbis 128 kbps sample and generated sine. Output at full volume. Internal clock multiplier $3.0\times$. TA=+25°C.

Parameter	Min	Typ	Max	Unit
Power Supply Consumption AVDD, Reset		0.6	5.0	μA
Power Supply Consumption CVDD = 1.8V, Reset		12	20.0	μA
Power Supply Consumption AVDD, sine test, 30 Ω + GBUF	30	36.9	60	mA
Power Supply Consumption CVDD = 1.8V, sine test	8	10	15	mA
Power Supply Consumption AVDD, no load		5		mA
Power Supply Consumption AVDD, output load 30 Ω		11		mA
Power Supply Consumption AVDD, 30 Ω + GBUF		11		mA
Power Supply Consumption CVDD = 1.8V		11		mA

4.5 Digital Characteristics

Parameter	Min	Max	Unit
High-Level Input Voltage (xRESET, XTALI, XTALO)	$0.7\times$IOVDD	IOVDD+0.3[1]	V
High-Level Input Voltage (other input pins)	$0.7\times$CVDD	IOVDD+0.3[1]	V
Low-Level Input Voltage	-0.2	$0.3\times$CVDD	V
High-Level Output Voltage at XTALO = -0.1 mA	$0.7\times$IOVDD		V
Low-Level Output Voltage at XTALO = 0.1 mA		$0.3\times$IOVDD	V
High-Level Output Voltage at I_O = -1.0 mA	$0.7\times$IOVDD		V
Low-Level Output Voltage at I_O = 1.0 mA		$0.3\times$IOVDD	V
Input Leakage Current	-1.0	1.0	μA
SPI Input Clock Frequency [2]		$\frac{CLKI}{7}$	MHz
Rise time of all output pins, load = 50 pF		50	ns

[1] Must not exceed 3.6V
[2] Value for SCI reads. SCI and SDI writes allow $\frac{CLKI}{4}$.

4.6 Switching Characteristics - Boot Initialization

Parameter	Symbol	Min	Max	Unit
XRESET active time		2		XTALI
XRESET inactive to software ready		22000	50000[1]	XTALI
Power on reset, rise time to CVDD		10		V/s

[1] DREQ rises when initialization is complete. You should not send any data or commands before that.

4.4 功耗

使用Ogg Vorbis格式128kbps样本及生成正弦波测试。满音量输出。内部时钟倍频器 3.0×，环境温度TA=+25°C。

参数	最小值	典型值	最大值	单位
电源功耗 AVDD，复位状态		0.6	5.0	微安 (μA)
电源功耗 CVDD=1.8V，复位状态		12	20.0	微安 (μA)
电源功耗 AVDD，正弦波测试，30Ω+ GBUF	30	36.9	60	mA
电源功耗 CVDD=1.8V，正弦波测试	8	10	15	mA
电源功耗 AVDD，空载		5		mA
电源功耗 AVDD，输出负载30Ω		11		mA
电源功耗 AVDD, 30Ω+ GBUF		11		mA
电源功耗 CVDD = 1.8V		11		mA

4.5 数字特性

参数	最小值	最大值	单位
高电平输入电压(xRESET, XTALI, XTALO)	$0.7\times$IOVDD	IOVDD+0.3[1]	V
高电平输入电压(其他输入引脚)	$0.7\times$CVDD	IOVDD+0.3[1]	V
低电平输入电压	-0.2	$0.3\times$CVDD	V
XTALO 高电平输出电压 = -0.1 mA	$0.7\times$IOVDD		V
XTALO 低电平输出电压 = 0.1 mA		$0.3\times$IOVDD	V
高电平输出电压 (I_O= -1.0 mA)	$0.7\times$IOVDD		V
低电平输出电压 (I_O= 1.0 mA)		$0.3\times$IOVDD	V
输入漏电流	-1.0	1.0	微安 (μA)
SPI 输入时钟频率 [2]		$\frac{CLKI}{7}$	兆赫兹
所有输出引脚上升时间，负载 = 50 pF		50	ns

[1] 不得超过 3.6V
[2] SCI 读取值。SCI 和 SDI 写入允许 $\frac{CLKI}{4}$.

4.6 开关特性 - 启动初始化

参数	符号	最小值	最大值	单位
XRESET 有效时间		2		XTALI
XRESET 无效至软件就绪		22000	50000[1]	XTALI
上电复位，上升至 CVDD 时间		10		V/s

[1] 初始化完成时 DREQ 上升。在此之前不应发送任何数据或命令。

5 Packages and Pin Descriptions

5.1 Packages

LPQFP-48 is a lead (Pb) free and also RoHS compliant package. RoHS is a short name of *Directive 2002/95/EC on the restriction of the use of certain hazardous substances in electrical and electronic equipment*.

5.1.1 LQFP-48

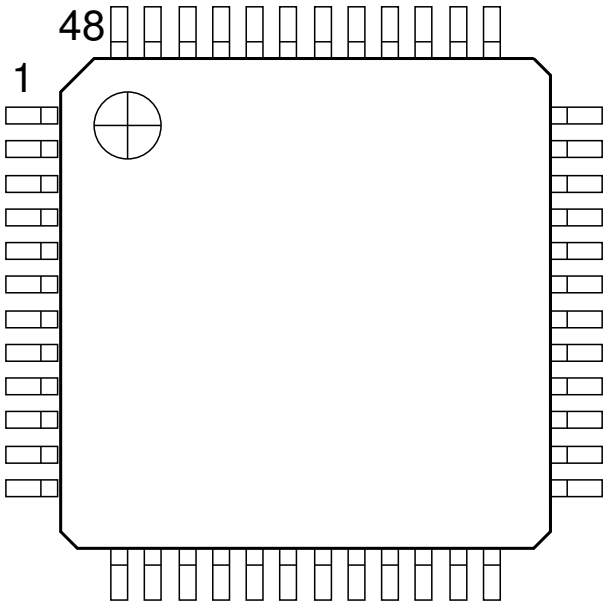


Figure 1: Pin configuration, LQFP-48.

LQFP-48 package dimensions are at *http://www.vlsi.fi/* .

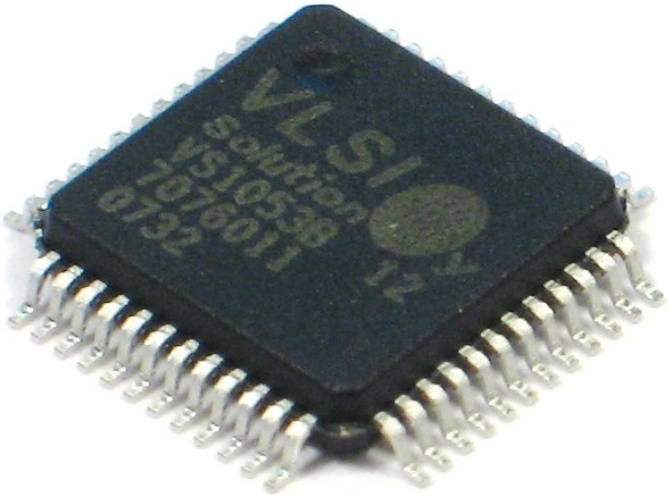


Figure 2: VS1053b in LQFP-48 packaging.

5 封装与引脚描述

5.1 封装形式

LQFP-48 是无铅且符合 RoHS 标准的封装。RoHS 是欧盟《关于限制在电子电气设备中使用某些有害物质的指令》(*2002/95/EC*) 的简称。

5.1.1 LQFP-48

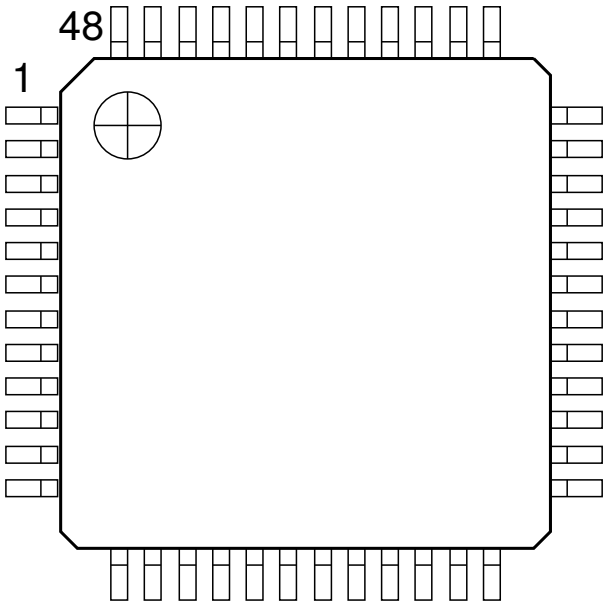


图1：引脚配置，LQFP-48封装。

LQFP-48封装尺寸详见*http://www.vlsi.fi/*。

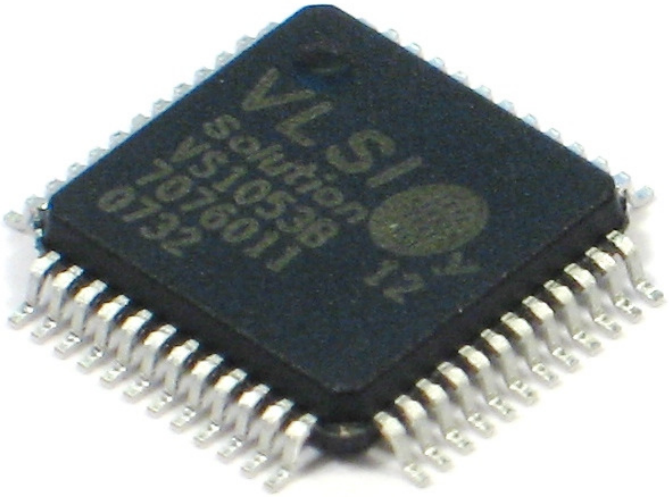


图2：采用LQFP-48封装的VS1053b芯片。

Pad Name	LQFP Pin	Pin Type	Function
MICP / LINE1	1	AI	Positive differential mic input, self-biasing / Line-in 1
MICN	2	AI	Negative differential mic input, self-biasing
XRESET	3	DI	Active low asynchronous reset, schmitt-trigger input
DGND0	4	DGND	Core & I/O ground
CVDD0	5	CPWR	Core power supply
IOVDD0	6	IOPWR	I/O power supply
CVDD1	7	CPWR	Core power supply
DREQ	8	DO	Data request, input bus
GPIO2 / DCLK[1]	9	DIO	General purpose IO 2 / serial input data bus clock
GPIO3 / SDATA[1]	10	DIO	General purpose IO 3 / serial data input
GPIO6 / I2S_SCLK[3]	11	DIO	General purpose IO 6 / I2S_SCLK
GPIO7 / I2S_SDATA[3]	12	DIO	General purpose IO 7 / I2S_SDATA
XDCS / BSYNC[1]	13	DI	Data chip select / byte sync
IOVDD1	14	IOPWR	I/O power supply
VCO	15	DO	For testing only (Clock VCO output)
DGND1	16	DGND	Core & I/O ground
XTALO	17	AO	Crystal output
XTALI	18	AI	Crystal input
IOVDD2	19	IOPWR	I/O power supply
DGND2	20	DGND	Core & I/O ground
DGND3	21	DGND	Core & I/O ground
DGND4	22	DGND	Core & I/O ground
XCS	23	DI	Chip select input (active low)
CVDD2	24	CPWR	Core power supply
GPIO5 / I2S_MCLK[3]	25	DIO	General purpose IO 5 / I2S_MCLK
RX	26	DI	UART receive, connect to IOVDD if not used
TX	27	DO	UART transmit
SCLK	28	DI	Clock for serial bus
SI	29	DI	Serial input
SO	30	DO3	Serial output
CVDD3	31	CPWR	Core power supply
XTEST	32	DI	Reserved for test, connect to IOVDD
GPIO0	33	DIO	Gen. purp. IO 0 (SPIBOOT), use 100 kΩ pull-down resistor[2]
GPIO1	34	DIO	General purpose IO 1
GND	35	DGND	I/O Ground
GPIO4 / I2S_LROUT[3]	36	DIO	General purpose IO 4 / I2S_LROUT
AGND0	37	APWR	Analog ground, low-noise reference
AVDD0	38	APWR	Analog power supply
RIGHT	39	AO	Right channel output
AGND1	40	APWR	Analog ground
AGND2	41	APWR	Analog ground
GBUF	42	AO	Common buffer for headphones, do NOT connect to ground!
AVDD1	43	APWR	Analog power supply
RCAP	44	AIO	Filtering capacitance for reference
AVDD2	45	APWR	Analog power supply
LEFT	46	AO	Left channel output
AGND3	47	APWR	Analog ground
LINE2	48	AI	Line-in 2 (right channel)

焊盘名称	LQFP 引脚	引脚类型	功能
MICP / LINE1	1	模拟输入	差分麦克风正极输入（自偏置）/ 线路输入1
MICN	2	模拟输入	差分麦克风负极输入（自偏置）
XRESET	3	数字输入	低电平有效异步复位，施密特触发器输入
DGND0	4	数字地	核心与I/O接地
CVDD0	5	内核电源	内核电源
IOVDD0	6	I/O电源	I/O电源
CVDD1	7	内核电源	内核电源
数据请求	8	数据输出	数据请求，输入总线
GPIO2 / DCLK[1]	9	数据输入	通用IO 2 / 串行输入数据总线时钟
GPIO3 / SDATA[1]	10	数据输入	通用IO 3 / 串行数据输入
GPIO6 / I2S_SCLK[3]	11	数据输入	通用IO 6 / I2S_SCLK
GPIO7 / I2S_SDATA[3]	12	数据输入	通用IO 7 / I2S_SDATA
XDCS / BSYNC[1]	13	数字输入	数据片选 / 字节同步
IO电源1	14	I/O电源	I/O电源
压控振荡器	15	数据输出	仅用于测试（时钟压控振荡器输出）
数字地1	16	数字地	核心与I/O接地
晶振输出	17	模拟输出	晶振输出
XTALI	18	模拟输入	晶振输入
IO电源2	19	I/O电源	I/O电源
数字地2	20	数字地	核心与I/O接地
数字地3	21	数字地	核心与I/O接地
数字地4	22	数字地	核心与I/O接地
片选信号	23	数字输入	片选输入（低电平有效）
CVDD2	24	内核电源	内核电源
GPIO5 / I2S_MCLK[3]	25	数据输入	通用输入输出端口5 / I2S主时钟
RX	26	数字输入	UART接收端，未使用时连接至IOVDD
TX	27	数据输出	UART发送端
SCLK	28	数字输入	串行总线时钟
SI	29	数字输入	串行输入
SO	30	DO3	串行输出
CVDD3	31	内核电源	内核电源
XTEST	32	数字输入	测试保留引脚，连接至IOVDD
GPIO0	33	数据输入	通用输入输出IO 0 (SPIBOOT)，使用 100 kΩ下拉电阻[2]
通用输入输出1	34	数据输入	通用输入输出接口1
地	35	数字地	输入/输出地
通用输入输出4 / I2S左右声道时钟输出[3]	36	数据输入	通用输入输出4 / I2S左右声道时钟输出
模拟地0	37	模拟电源	模拟地，低噪声参考基准
模拟电源0	38	模拟电源	模拟电源供电
右声道	39	模拟输出	右声道输出
模拟地1	40	模拟电源	模拟地
AGND2	41	模拟电源	模拟地
GBUF	42	模拟输出	耳机公共缓冲器，切勿接地！
AVDD1	43	模拟电源	模拟电源供电
RCAP	44	AIO	基准电压滤波电容
AVDD2	45	模拟电源	模拟电源供电
LEFT	46	模拟输出	左声道输出
AGND3	47	模拟电源	模拟地
LINE2	48	模拟输入	线路输入2（右声道）

[1] First pin function is active in New Mode, latter in Compatibility Mode.

[2] Unless pull-down resistor is used, SPI Boot is tried. See Chapter 10.9 for details.

[3] If I2S_CF_ENA is '0' the pins are used for GPIO. See Chapter 11.14 for details.

Pin types:

Type	Description
DI	Digital input, CMOS Input Pad
DO	Digital output, CMOS Input Pad
DIO	Digital input/output
DO3	Digital output, CMOS Tri-stated Output Pad
AI	Analog input

Type	Description
AO	Analog output
AIO	Analog input/output
APWR	Analog power supply pin
DGND	Core or I/O ground pin
CPWR	Core power supply pin
IOPWR	I/O power supply pin

[1] 首引脚功能在新模式下激活，兼容模式下激活次引脚功能
[2] 除非使用下拉电阻，否则将尝试SPI引导详见章节10.9
[3] 若I2S_CF_ENA为'0'，则引脚用作GPIO。详见章节11.14

引脚类型：

类型	描述
数字输入	数字输入，CMOS输入焊盘
数据输出	数字输出，CMOS输入焊盘
数据输入	数字输入/输出
DO3	数字输出，CMOS三态输出 焊盘
模拟输入	模拟输入

类型	描述
模拟输出	模拟输出
AIO	模拟输入/输出
模拟电源	模拟电源引脚
数字地	核心或I/O接地引脚
内核电源	核心电源引脚
I/O电源	I/O电源引脚

6 Connection Diagram, LQFP-48

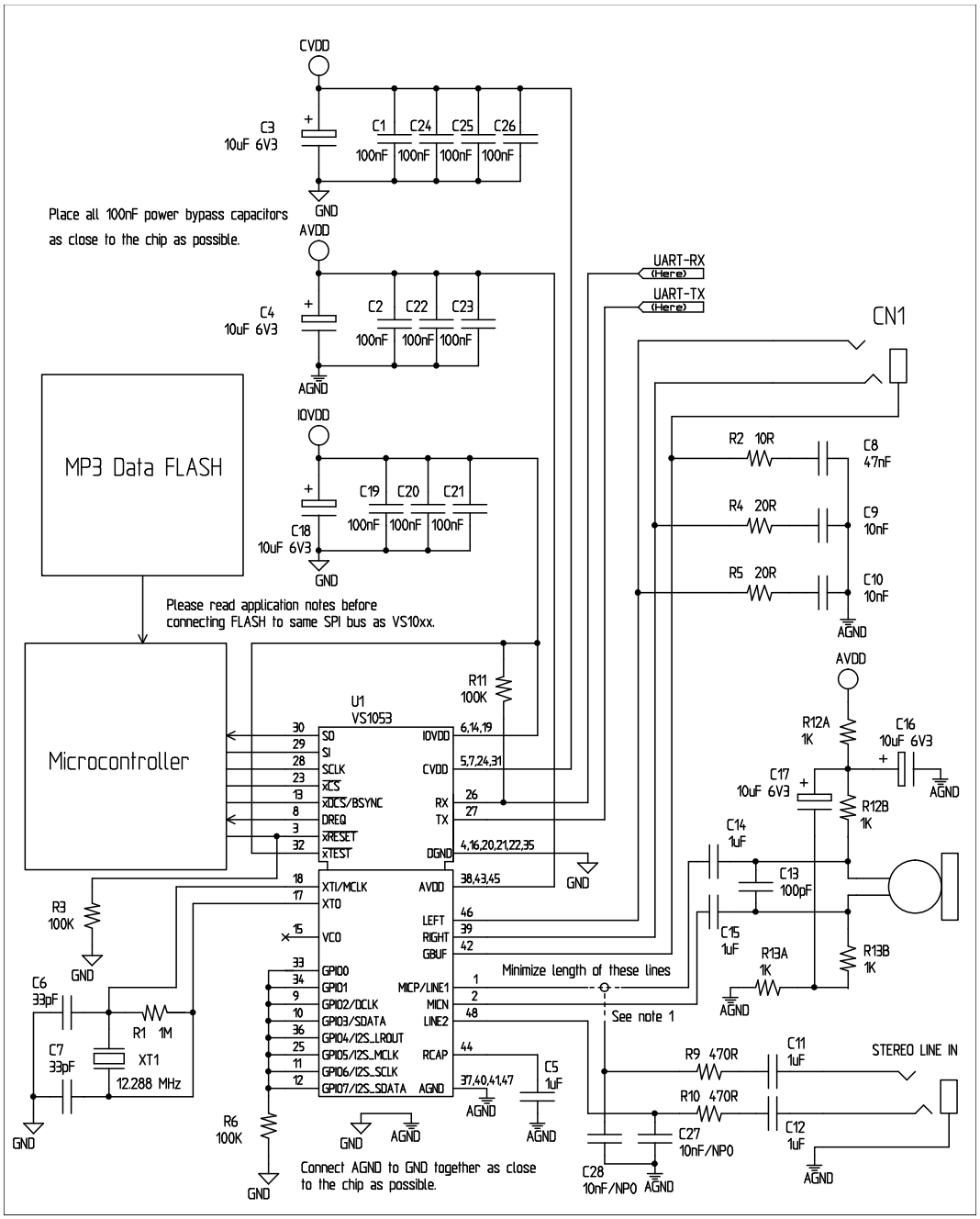


Figure 3: Typical connection diagram using LQFP-48.

Figure 3 shows a typical connection diagram for VS1053.

Figure Note 1: Connect either Microphone In or Line In, but not both at the same time.

Note: This connection assumes SM_SDINEW is active (see Chapter 9.6.1). If also SM_SDISHARE is used, xDCS should be tied low or high (see Chapter 7.1.1).

图6：LQFP-48封装连接图

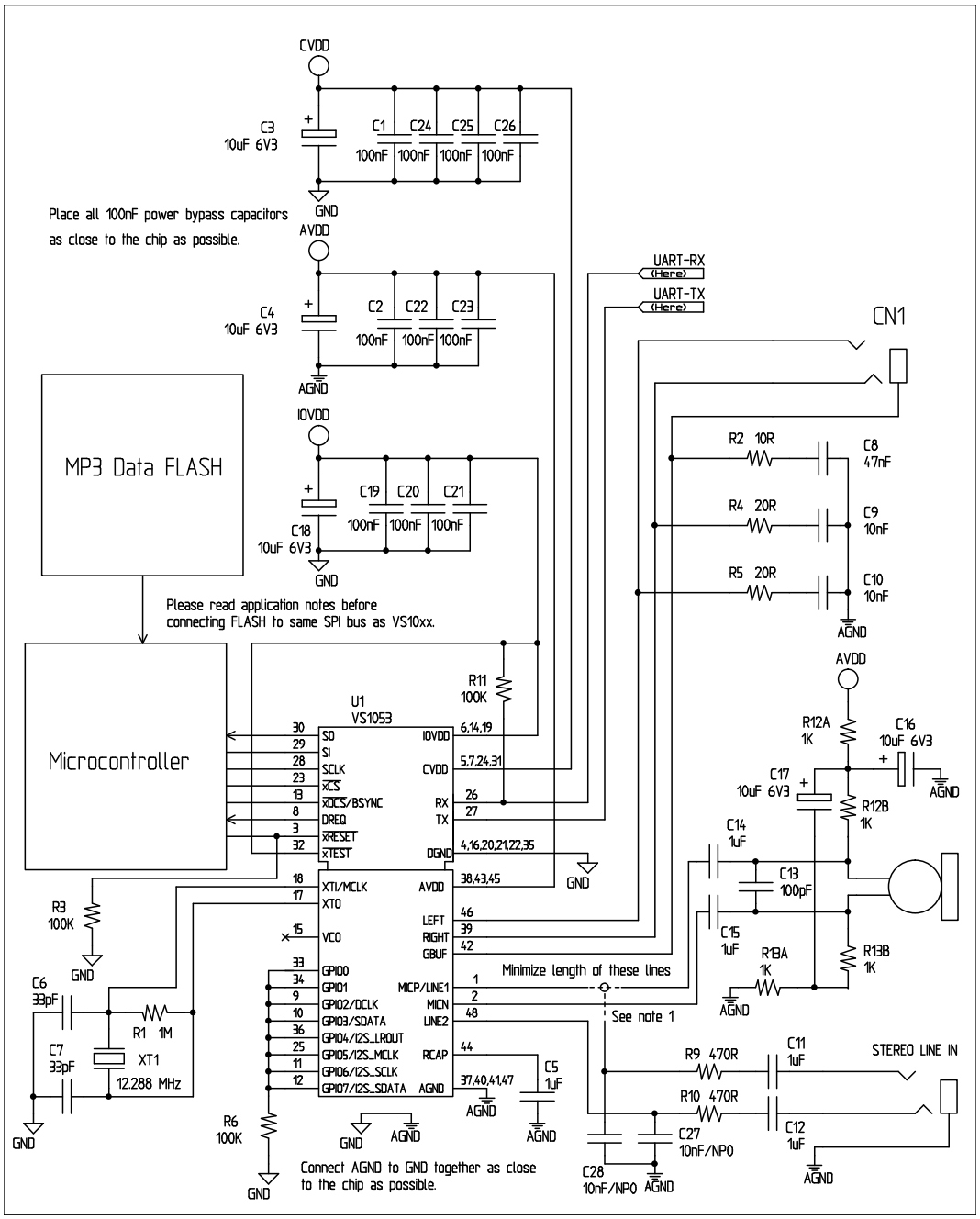


图3：采用LQFP-48封装的典型连接图

图3展示了VS1053的典型连接示意图

图注1：麦克风输入与线路输入仅可连接其中一路，不可同时使用

注意：此连接方式需确保SM_SDINEW有效（参见第9.6.1章）。若同时使用SM_SDISHARE功能，xDCS引脚应上拉或下拉（参见第7.1.1章）

The common buffer GBUF can be used for common voltage (1.23 V) for earphones. This will eliminate the need for large isolation capacitors on line outputs, and thus the audio output pins from VS1053b may be connected directly to the earphone connector.

GBUF must NOT be connected to ground under any circumstances. If GBUF is not used, LEFT and RIGHT must be provided with coupling capacitors. To keep GBUF stable, you should always have the resistor and capacitor even when GBUF is not used. See application notes for details.

Unused GPIO pins should have a pull-down resistor. Unused line and microphone inputs should not be connected.

If UART is not used, RX should be connected to IOVDD and TX be unconnected.

Do not connect any external load to XTALO.

公共缓冲器GBUF可用作耳机的公共参考电压(1.23V)。 这将无需在线路输出端配备大型隔直电容，因此VS1053b的音频输出引脚可直接连接至耳机接口。

在任何情况下都绝对不可将GBUF接地。若不使用GBUF，则左/右声道必须配备耦合电容。为保持GBUF稳定，即使未使用也应保留电阻和电容元件。详见应用笔记说明。

未使用的GPIO引脚应配置下拉电阻。未使用的线路输入和麦克风输入应保持悬空。

若未使用UART，RX引脚应连接至IOVDD，TX引脚保持悬空。

切勿在XTALO引脚连接任何外部负载。

7 SPI Buses

The SPI Bus - which was originally used in some Motorola devices - has been used for both VS1053b's Serial Data Interface SDI (Chapters 7.3 and 9.4) and Serial Control Interface SCI (Chapters 7.4 and 9.5).

7.1 SPI Bus Pin Descriptions

7.1.1 VS10xx Native Modes (New Mode, recommended)

These modes are active on VS1053b when SM_SDINEW is set to 1 (default at startup). DCLK and SDATA are not used for data transfer and they can be used as general-purpose I/O pins (GPIO2 and GPIO3). BSYNC function changes to data interface chip select (XDCS).

SDI Pin	SCI Pin	Description
XDCS	XCS	Active low chip select input. A high level forces the serial interface into standby mode, ending the current operation. A high level also forces serial output (SO) to high impedance state. If SM_SDISHARE is 1, pin XDCS is not used, but the signal is generated internally by inverting XCS.
SCK		Serial clock input. The serial clock is also used internally as the master clock for the register interface. SCK can be gated or continuous. In either case, the first rising clock edge after XCS has gone low marks the first bit to be written.
SI		Serial input. If a chip select is active, SI is sampled on the rising CLK edge.
-	SO	Serial output. In reads, data is shifted out on the falling SCK edge. In writes SO is at a high impedance state.

7.1.2 VS1001 Compatibility Mode (deprecated, do not use in new designs)

This mode is active when SM_SDINEW is set to 0. In this mode, DCLK, SDATA and BSYNC are active.

SDI Pin	SCI Pin	Description
-	XCS	Active low chip select input. A high level forces the serial interface into standby mode, ending the current operation. A high level also forces serial output (SO) to high impedance state.
BSYNC	-	SDI data is synchronized with a rising edge of BSYNC.
DCLK	SCK	Serial clock input. The serial clock is also used internally as the master clock for the register interface. SCK can be gated or continuous. In either case, the first rising clock edge after XCS has gone low marks the first bit to be written.
SDATA	SI	Serial input. SI is sampled on the rising SCK edge, if XCS is low.
-	SO	Serial output. In reads, data is shifted out on the falling SCK edge. In writes SO is at a high impedance state.

7组SPI总线

最初应用于摩托罗拉设备的SPI总线，在VS1053b中同时用于串行数据接口SDI（第7.3章与9.4章）和串行控制接口SCI（第7.4章与9.5章）。

7.1 SPI总线引脚说明

7.1.1 VS10xx原生模式（新模式，推荐）

当SM_SDINEW设置为1（启动默认值）时，VS1053b将激活这些模式。DCLK和SDATA不用于数据传输，可作为通用I/O引脚（GPIO2和GPIO3）。BSYNC功能转变为数据接口片选信号（XDCS）。

SDI引脚	SCI引脚	描述
XDCS	片选信号	低电平有效的片选输入信号。高电平将强制串行接口进入待机模式，终止当前操作。高电平同时强制串行输出（SO）进入高阻态。若SM_SDISHARE为1，则XDCS引脚不启用，该信号由XCS反相后内部生成。
SCK		串行时钟输入该时钟同时作为寄存器接口的主时钟内部使用。 SCK可采用门控模式或连续时钟模式。无论何种情况，XCS变为低电平后的首个时钟上升沿标志待写入的首个数据位开始。
SI		串行输入。片选信号有效时，SI在CLK上升沿被采样。
-	SO	串行输出。读取操作中，数据在SCK下降沿移出。 写入操作时SO处于高阻态。

7.1.2 VS1001兼容模式（已弃用，新设计请勿使用）

当SM_SDINEW设为0时激活此模式。此模式下DCLK、SDATA和BSYNC信号有效。

SDI引脚	SCI引脚	描述
-	片选信号	低电平有效的片选输入信号。高电平将强制串行接口进入待机模式，终止当前操作。高电平同时强制串行输出(SO)进入高阻态。
特定术语：BSYNC	-	SDI数据在BSYNC上升沿同步。
DCLK	SCK	串行时钟输入该时钟同时作为寄存器接口的主时钟内部使用。 SCK可采用门控模式或连续时钟模式。无论何种情况，XCS变为低电平后的首个时钟上升沿标志待写入的首个数据位开始。
SDATA	SI	串行输入。若XCS为低电平，SI在SCK上升沿被采样。
-	SO	串行输出。读取操作中，数据在SCK下降沿移出。 写入操作时SO处于高阻态。

7.2 Data Request Pin DREQ

The DREQ pin/signal is used to signal if VS1053b's 2048-byte FIFO is capable of receiving data. If DREQ is high, VS1053b can take at least 32 bytes of SDI data or one SCI command. DREQ is turned low when the stream buffer is too full and for the duration of an SCI command.

Because of the 32-byte safety area, the sender may send upto 32 bytes of SDI data at a time without checking the status of DREQ, making controlling VS1053b easier for low-speed microcontrollers.

Note: DREQ may turn low or high at any time, even during a byte transmission. Thus, DREQ should only be used to decide whether to send more bytes. A transmission that has already started doesn't need to be aborted.

Note: In VS1053b DREQ also goes down while an SCI operation is in progress.

There are cases when you still want to send SCI commands when DREQ is low. Because DREQ is shared between SDI and SCI, you can not determine if an SCI command has been executed if SDI is not ready to receive data. In this case you need a long enough delay after every SCI command to make certain none of them are missed. The SCI Registers table in Chapter 9.6 gives the worst-case handling time for each SCI register write.

Note: The status of DREQ can also be read through SCI with the following code. For details on SCI registers, see Chapter 7.4.

```
// This example reads status of DREQ pin through the SPI/SCI register
// interface.
#define SCI_WRAMADDR 7
#define SCI_WRAM 6
while (!endOfFile) {
  int dreq;
  WriteSciReg(SCI_WRAMADDR, 0xC012); // Send address of DREQ register
  dreq = ReadSciReg(SCI_WRAM) & 1;   // Read value of DREQ (in bit 0)
  if (dreq) {
    // DREQ high: send 1-32 bytes audio data
  } else {
    // DREQ low: wait 5 milliseconds (so that VS10xx doesn't get
    // continuous SCI operations)
  }
} /* while (!endOfFile) */
```

7.2 数据请求引脚DREQ

DREQ引脚/信号用于指示VS1053b的2048字节FIFO是否具备数据接收能力。若DREQ为高电平，VS1053b可接收至少32字节的SDI数据或一条SCI命令。
当流缓冲区过满或在SCI命令执行期间，DREQ将变为低电平。

由于存在32字节安全区，发送方可无需检查DREQ状态直接发送最多32字节SDI数据，这使得低速微控制器更易控制VS1053b。

注意：DREQ可能在任何时刻（包括字节传输过程中）变为低电平或高电平。因此，DREQ仅应用于判断是否继续发送字节。已启动的传输过程无需中止。

注意：在VS1053b中，SCI操作进行期间DREQ同样会变为低电平。

某些情况下仍需在DREQ低电平时发送SCI命令。由于DREQ由SDI和SCI共享，若SDI尚未准备好接收数据，则无法确定SCI命令是否已执行。此种情况下，每次执行SCI命令后需预留足够长的延迟，以确保所有指令均被正确处理。第9.6章的SCI寄存器表列出了各SCI寄存器写入操作的最坏情况处理时间。

注意：DREQ状态也可通过以下SCI代码读取。关于SCI寄存器的详细信息，请参阅第7.4章。

```
// 此示例通过SPI/SCI寄存器接口读取DREQ引脚状态
// 接口。
#define SCI_WRAMADDR 7
#define SCI_WRAM 6
while (!endOfFile) {
   int dreq;
   WriteSciReg(SCI_WRAMADDR, 0xC012); // 发送DREQ寄存器地址
   dreq = ReadSciReg(SCI_WRAM) & 1;       // 读取DREQ值（位于位0）
   if (dreq) {
      // DREQ高电平：发送1-32字节音频数据
   } else {
      // DREQ低电平：等待5毫秒（防止VS10xx芯片
      // 连续SCI操作）
   }
} /* while (!文件结束) */
```

7.3 Serial Protocol for Serial Data Interface (SPI / SDI)

The serial data interface operates in slave mode so DCLK signal must be generated by an external circuit.

Data (SDATA signal) can be clocked in at either the rising or falling edge of DCLK (Chapter 9.6).

VS1053b assumes its data input to be byte-sychronized. SDI bytes may be transmitted either MSb or LSb first, depending of register SCI_MODE bit SM_SDIORD (Chapter 9.6.1).

The firmware is able to accept the maximum bitrate the SDI supports.

7.3.1 SDI in VS10xx Native Modes (New Mode, recommended)

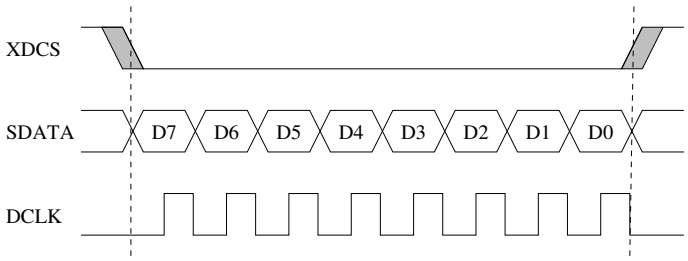


Figure 4: SDI in VS10xx Native Mode, single-byte transfer

In VS10xx native modes (SM_NEWMODE is 1), byte synchronization is achieved by XDCS, as shown in Figure 4. The state of XDCS may not change while a data byte transfer is in progress. XDCS does not need to be deactivated and reactivated for every byte transfer, as shown in Figure 5. However, to maintain data synchronization even if there are occasional clock glitches, it is recommended to deactivate and reactivate XDCS every now and then, for example after each 32 bytes of data.

Note that when sending data through SDI you have to check the Data Request Pin DREQ at least after every 32 bytes (Chapter 7.2).

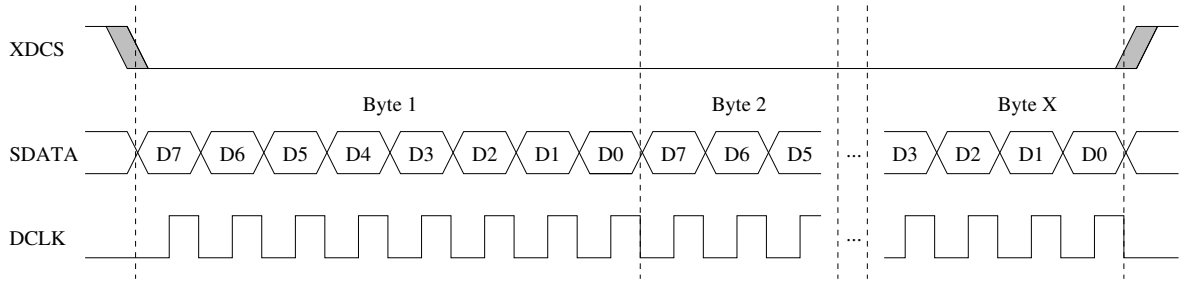


Figure 5: SDI in VS10xx Native Mode, multi-byte transfer, $X \geq 1$

If SM_SDISHARE is 1, the XDCS signal is internally generated by inverting the XCS input.

7.3 串行数据接口协议（SPI/SDI）

串行数据接口工作于从模式，因此DCLK信号必须由外部电路产生。

数据（SDATA信号）可在DCLK的上升沿或下降沿输入（第9.6章）。

VS1053b要求其数据输入为字节同步。SDI字节传输可采用最高位优先(MSb)或最低位优先(LSb)，具体取决于SCI_MODE寄存器的SM_SDIORD位（第9.6.1章）。

固件可支持SDI接口的最大比特率传输。

7.3.1 VS10xx原生模式下的SDI（新模式，推荐）

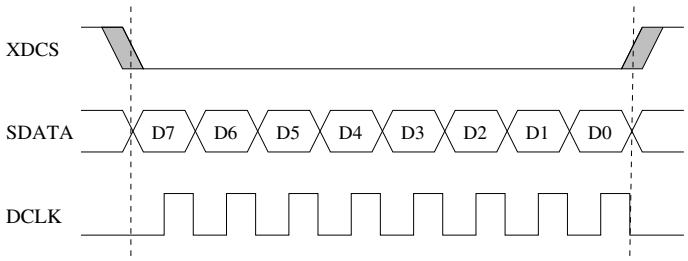


图4：VS10xx原生模式下的SDI单字节传输

在VS10xx系列原生模式下（SM_NEWMODE为1时），字节同步通过XDCS实现，如图4所示。数据传输过程中XDCS引脚状态不应改变。
如图5所示，无需在每次字节传输时重新置位XDCS信号。但为维持数据同步（应对偶发的时钟干扰），建议定期重新置位XDCS信号，例如每传输32字节后执行一次。

注意：通过SDI发送数据时，必须至少每32字节检查一次数据请求引脚DREQ状态（详见章节7.2）。

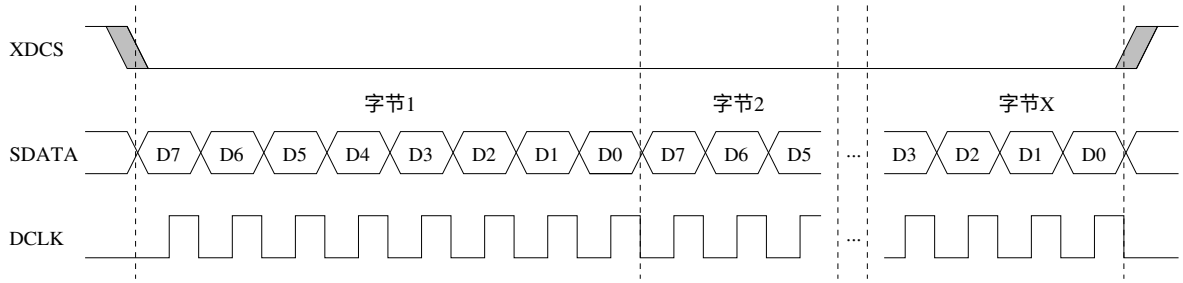


图5：VS10xx原生模式下的SDI多字节传输（$X \geq 1$）

若寄存器SM_SDISHARE设为1，则XDCS信号由XCS输入信号经内部反相生成。

7.3.2 SDI Timing Diagram in VS10xx Native Modes (New Mode)

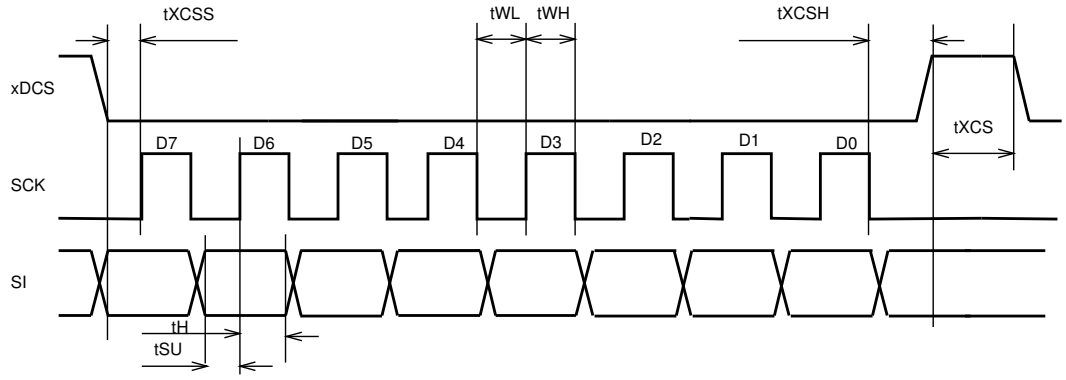


Figure 6: SDI timing diagram

Figure 6 presents SDI bus timing.

Symbol	Min	Max	Unit
tXCSS	5		ns
tSU	0		ns
tH	2		CLKI cycles
tWL	2		CLKI cycles
tWH	2		CLKI cycles
tXCSH	1		CLKI cycles
tXCS	0		CLKI cycles

Note: xDCS is not required to go high between bytes, so tXCS is 0.

Note: Although the timing is derived from the internal clock CLKI, the system always starts up in $1.0\times$ mode, thus CLKI=XTALI. After you have configured a higher clock through SCI_CLOCKF and waited for DREQ to rise, you can use a higher SPI speed as well.

7.3.2 VS10xx 原生模式下的SDI时序图（新模式）

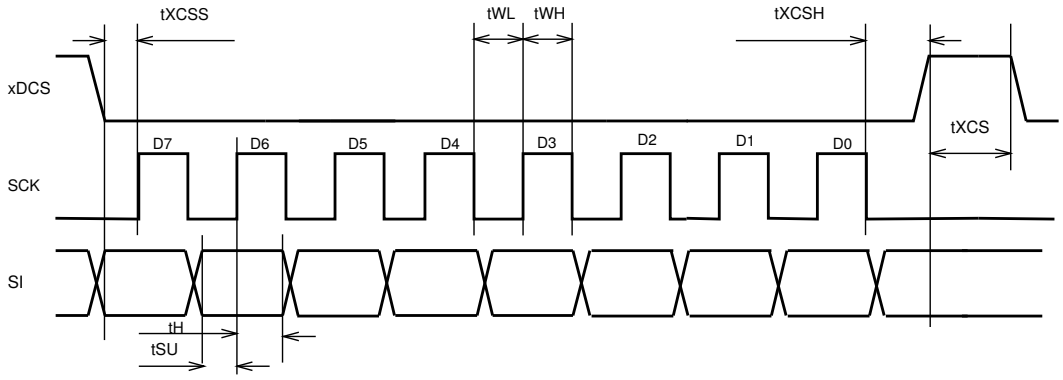


图6：SDI时序图

图6展示了SDI总线时序

符号	最小值	最大值	单位
tXCSS	5		ns
tSU	0		ns
tH	2		特定术语：CLKI 时钟周期
tWL	2		特定术语：CLKI 时钟周期
tWH	2		特定术语：CLKI 时钟周期
tXCSH	1		特定术语：CLKI 时钟周期
tXCS	0		特定术语：CLKI 时钟周期

注：xDCS在字节传输间无需置高，因此tXCS为0

注：虽然时序源自内部时钟CLKI，但系统始终以$1.0\times$模式启动，此时CLKI=XTALI。通过SCI_CLOCKF寄存器配置更高时钟频率并等待DREQ信号上升后，即可使用更高SPI传输速率

7.3.3 SDI in VS1001 Compatibility Mode (deprecated, do not use in new designs)

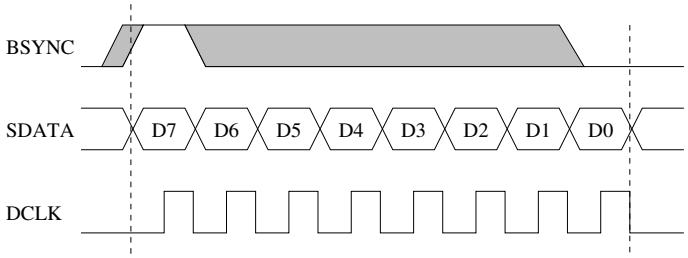


Figure 7: SDI in VS1001 Mode - one byte transfer. Do not use in new designs!

When VS1053b is running in VS1001 compatibility mode, a BSYNC signal must be generated to ensure correct bit-alignment of the input bitstream, as shown in Figures 7 and 8.

The first DCLK sampling edge (rising or falling, depending on selected polarity), during which the BSYNC is high, marks the first bit of a byte (LSB, if LSB-first order is used, MSB, if MSB-first order is used). If BSYNC is '1' when the last bit is received, the receiver stays active and next 8 bits are also received.

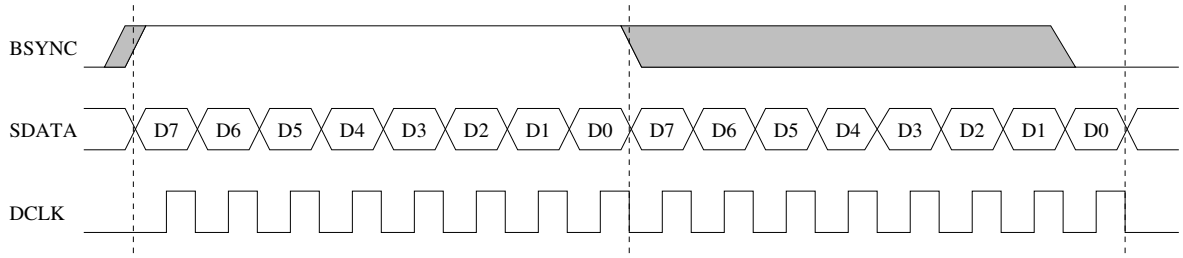


Figure 8: SDI in VS1001 Mode - two byte transfer. Do not use in new designs!

7.3.4 Passive SDI Mode (deprecated, do not use in new designs)

If SM_NEWMODE is 0 and SM_SDISHARE is 1, the operation is otherwise like the VS1001 compatibility mode, but bits are only received while the BSYNC signal is '1'. Rising edge of BSYNC is still used for synchronization.

7.3.3 VS1001兼容模式下的SDI（已弃用，新设计请勿使用）

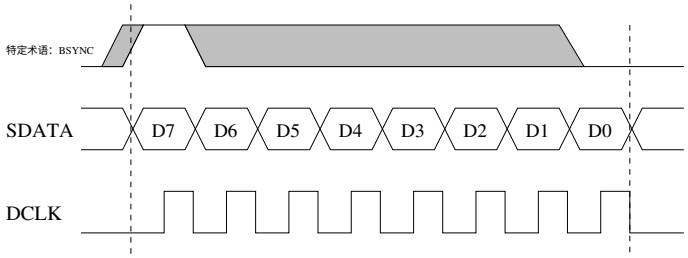


图7：VS1001模式下的SDI传输——单字节传输示意图请勿用于新设计！

当VS1053b运行在VS1001兼容模式时，必须生成BSYNC信号以确保输入比特流的正确比特对齐，如图7和图8所示。

在BSYNC为高电平期间，首个DCLK采样边沿（上升沿或下降沿，取决于所选极性）标记字节的首个比特（若采用LSB在前顺序则为最低有效位，若采用MSB在前顺序则为最高有效位）。若接收最后一位比特时BSYNC为'1'，则接收器保持激活状态，并继续接收后续8位比特。

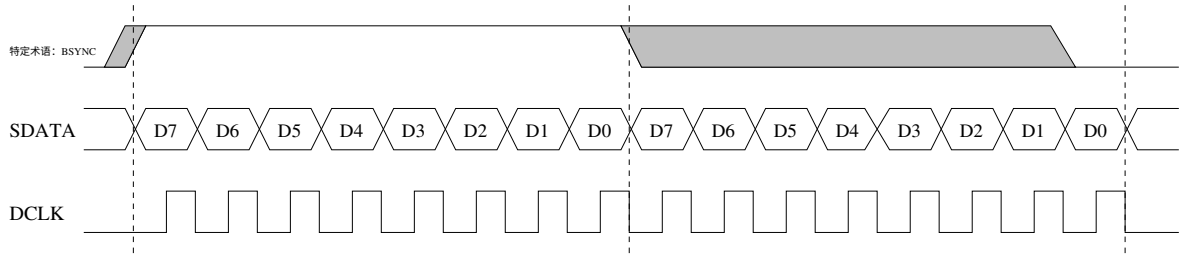


图8：VS1001模式下的SDI传输 - 双字节传输请勿用于新设计！

7.3.4 被动SDI模式（已弃用，请勿用于新设计）

若SM_NEWMODE为0且SM_SDISHARE为1，其操作方式类似VS1001兼容模式，但仅在BSYNC信号为'1'时接收比特。BSYNC的上升沿仍用于同步。

7.4 Serial Protocol for Serial Command Interface (SPI / SCI)

The serial bus protocol for the Serial Command Interface SCI (Chapter 9.5) consists of an instruction byte, address byte and one 16-bit data word. Each read or write operation can read or write a single register. Data bits are read at the rising edge, so the user should update data at the falling edge. Bytes are always send MSb first. XCS should be low for the full duration of the operation, but you can have pauses between bits if needed.

The operation is specified by an 8-bit instruction opcode. The supported instructions are read and write. See table below.

Instruction		
Name	Opcode	Operation
READ	0b0000 0011	Read data
WRITE	0b0000 0010	Write data

Note: VS1053b sets DREQ low after each SCI operation. The duration depends on the operation. It is not allowed to finish a new SCI/SDI operation before DREQ is high again.

7.4.1 SCI Read

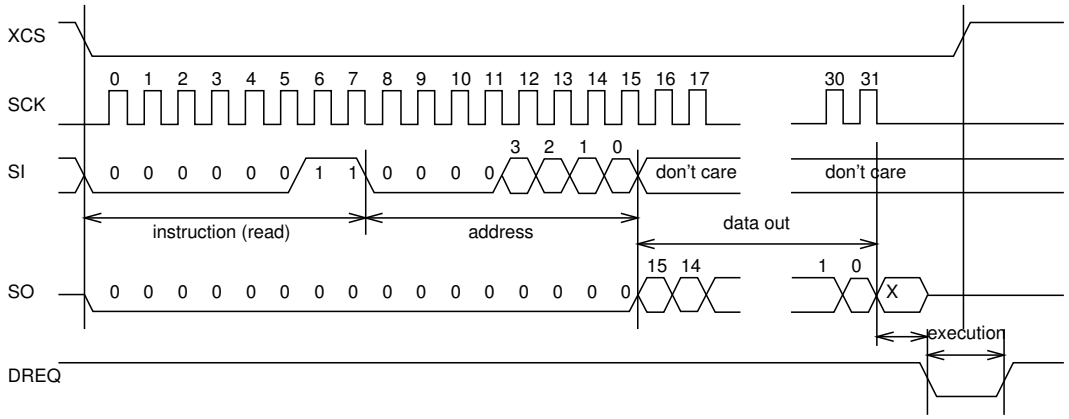


Figure 9: SCI word read

VS1053b registers are read from using the following sequence, as shown in Figure 9. First, XCS line is pulled low to select the device. Then the READ opcode (0x3) is transmitted via the SI line followed by an 8-bit word address. After the address has been read in, any further data on SI is ignored by the chip. The 16-bit data corresponding to the received address will be shifted out onto the SO line.

XCS should be driven high after data has been shifted out.

DREQ is driven low for a short while when in a read operation by the chip. This is a very short time and doesn't require special user attention.

7.4 串行命令接口串行协议 (SPI/SCI)

串行命令接口SCI（第9.5章）的总线协议包含一个指令字节、一个地址字节和一个16位数据字。每次读/写操作可读取或写入单个寄存器。数据位在上升沿被读取，因此用户应在下降沿更新数据。字节始终以最高位(MSb)优先发送。XCS在整个操作期间应保持低电平，但必要时可在位传输间插入暂停。

操作由8位指令操作码指定。支持的指令包括读取和写入，详见下表。

指令		
名称	操作码	操作
读取	0b0000 0011	读取数据
写入	0b0000 0010	写入数据

注意：VS1053b在每次SCI操作后会将DREQ置为低电平。持续时间取决于具体操作。在DREQ再次变高之前，不允许完成新的SCI/SDI操作。

7.4.1 SCI读取

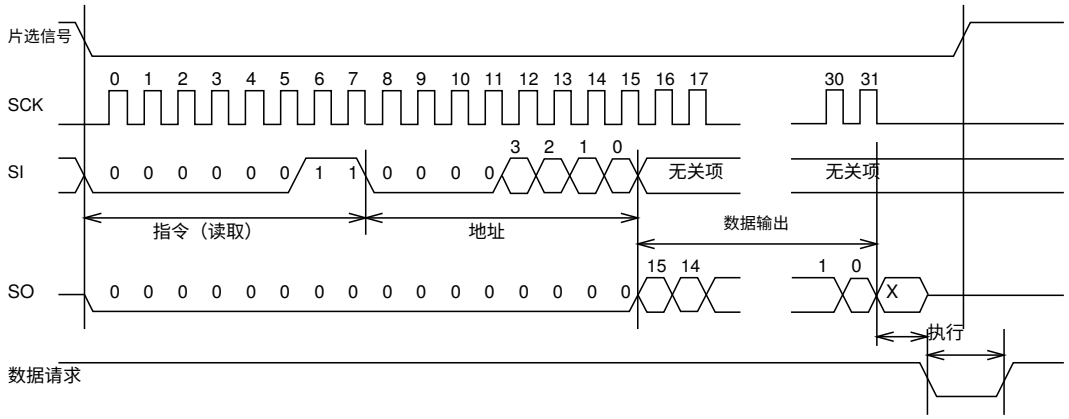


图9：SCI字读取时序

如图9所示，读取VS1053b寄存器需遵循以下时序序列。首先拉低XCS线路以选中器件。随后通过SI线路传输读取操作码(0x3)，后接8-bit word address。 地址读取完成后，芯片将忽略SI线上的后续数据。对应接收地址的16位数据将通过SO线路移出。

数据移出后应将XCS置为高电平。

芯片执行读取操作时，DREQ会短暂保持低电平状态。这段时间非常短暂，无需用户特别关注。

7.4.2 SCI Write

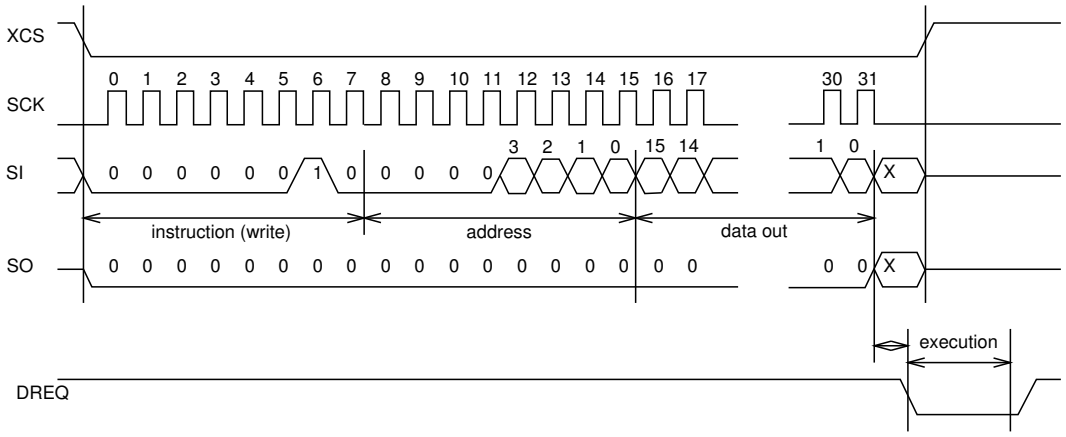


Figure 10: SCI word write

VS1053b registers are written from using the following sequence, as shown in Figure 10. First, XCS line is pulled low to select the device. Then the WRITE opcode (0x2) is transmitted via the SI line followed by an 8-bit word address.

After the word has been shifted in and the last clock has been sent, XCS should be pulled high to end the WRITE sequence.

After the last bit has been sent, DREQ is driven low for the duration of the register update, marked “execution” in the figure. The time varies depending on the register and its contents (see table in Chapter 9.6 for details). If the maximum time is longer than what it takes from the microcontroller to feed the next SCI command or SDI byte, status of DREQ must be checked before finishing the next SCI/SDI operation.

7.4.3 SCI Multiple Write

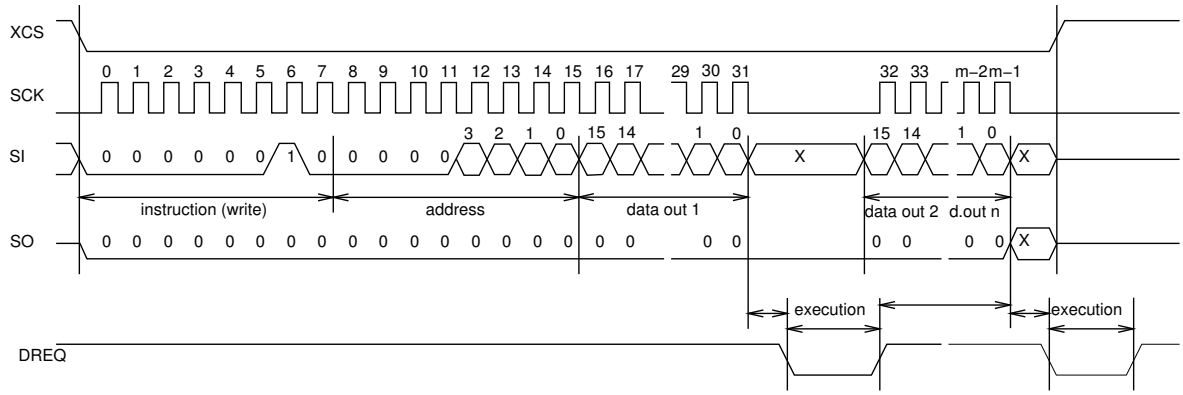


Figure 11: SCI multiple word write

VS1053b allows for the user to send multiple words to the same SCI register, which allows fast SCI uploads, shown in Figure 11. The main difference to a single write is that instead of

7.4.2 SCI写操作

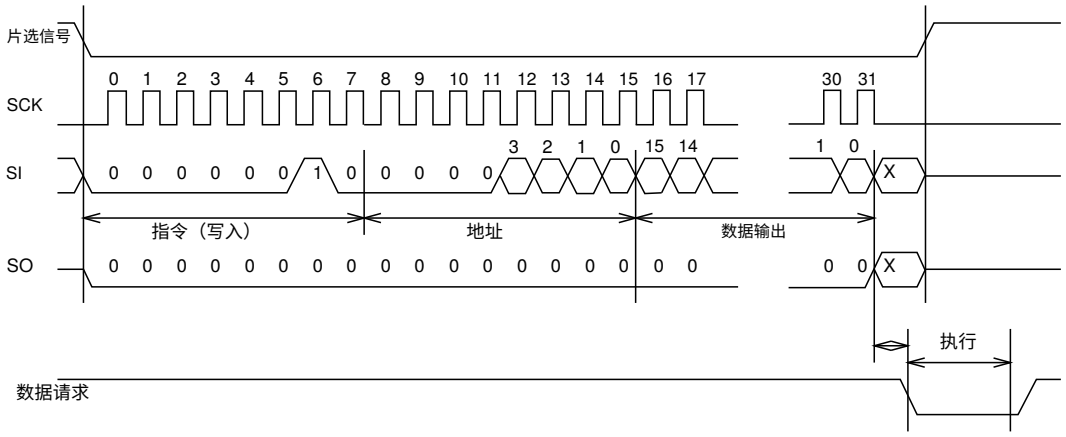


图10：SCI字写入时序

如图10所示，VS1053b寄存器通过以下时序进行写入操作。首先拉低XCS线路以选中器件。随后通过SI线路传输写操作码(0x2)，紧接着发送8位字地址。

待所有字数据移入且最后一个时钟信号发送完毕后，应将XCS拉高以结束写入时序。

末位数据发送完成后，DREQ将在寄存器更新期间（图中标记为'执行期'）保持低电平。该时间长短取决于寄存器及其内容（详见第9.6章表格说明）。 若最大耗时超过微控制器输入下一条SCI指令或SDI字节所需时间，则必须在完成后续SCI/SDI操作前检查DREQ状态。

7.4.3 SCI多字写入

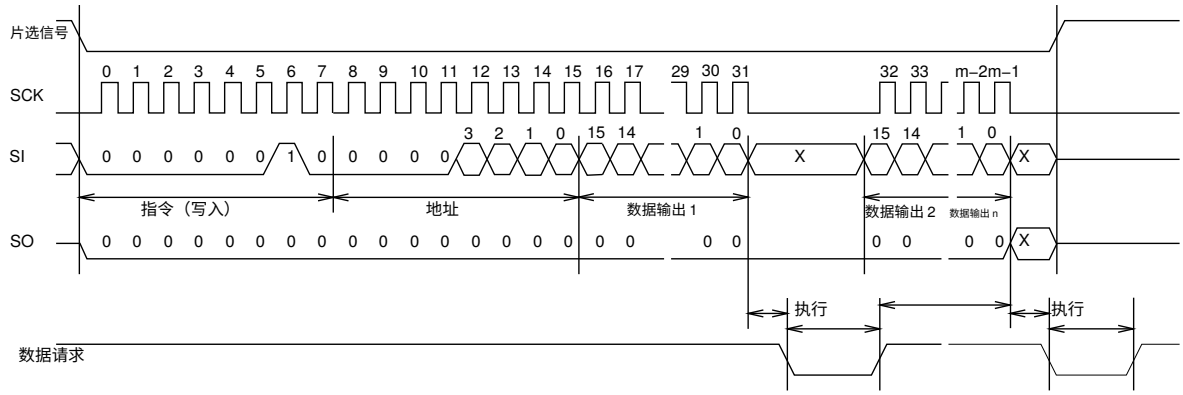


图11：SCI多字写入时序

VS1053b允许用户向同一SCI寄存器发送多个数据字，从而实现快速SCI上传，如图11所示。与单字写入的主要区别在于：

bringing XCS up after sending the last bit of a data word, the next data word is sent immediately. After the last data word, XCS is driven high as with a single word write.

After the last bit of a word has been sent, DREQ is driven low for the duration of the register update, marked “execution” in the figure. The time varies depending on the register and its contents (see table in Chapter 9.6 for details). If the maximum time is longer than what it takes from the microcontroller to feed the next SCI command or SDI byte, status of DREQ must be checked before finishing the next SCI/SDI operation.

7.4.4 SCI Timing Diagram

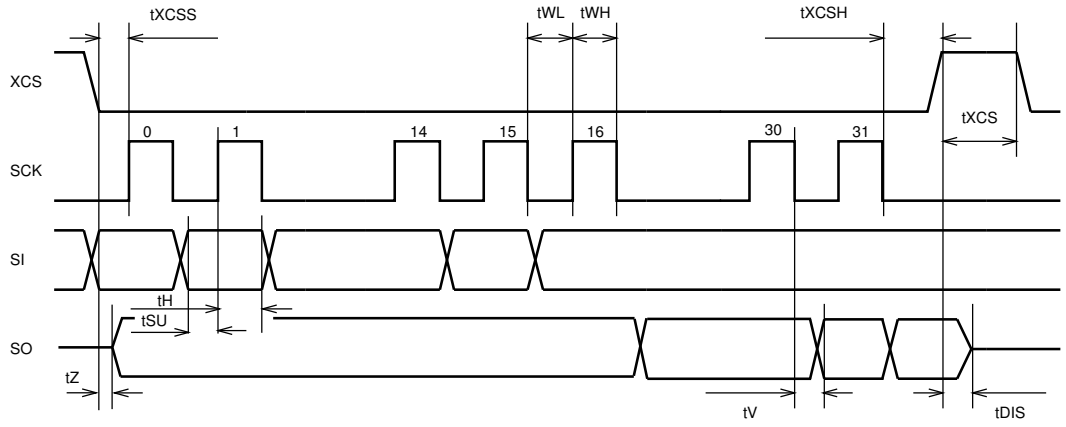


Figure 12: SPI timing diagram

The SCI timing diagram is presented in Figure 12.

Symbol	Min	Max	Unit
tXCSS	5		ns
tSU	0		ns
tH	2		CLKI cycles
tZ	0		ns
tWL	2		CLKI cycles
tWH	2		CLKI cycles
tV	2 (+ 25 ns[1])		CLKI cycles
tXCSH	1		CLKI cycles
tXCS	2		CLKI cycles
tDIS		10	ns

[1] 25 ns is when pin loaded with 100 pF capacitance. The time is shorter with lower capacitance.

Note: Although the timing is derived from the internal clock CLKI, the system always starts up in $1.0\times$ mode, thus CLKI=XTALI. After you have configured a higher clock through SCI_CLOCKF and waited for DREQ to rise, you can use a higher SPI speed as well.

Note: Because tWL + tWH + tH is $6\times$CLKI + 25 ns, the maximum speed for SCI reads is CLKI/7.

发送完某数据字的最后一位后不拉高XCS，而是立即发送下一个数据字。
最后一个数据字发送完毕后，XCS的拉高操作与单字写入相同。

某数据字最后一位发送完成后，DREQ将在寄存器更新期间保持低电平（图中标注为“执行阶段”）。 该时间长短取决于寄存器及其内容（详见第9.6章表格说明）。 若最大耗时超过微控制器输入下一条SCI指令或SDI字节所需时间，则必须在完成后续SCI/SDI操作前检查DREQ状态。

7.4.4 SCI时序图

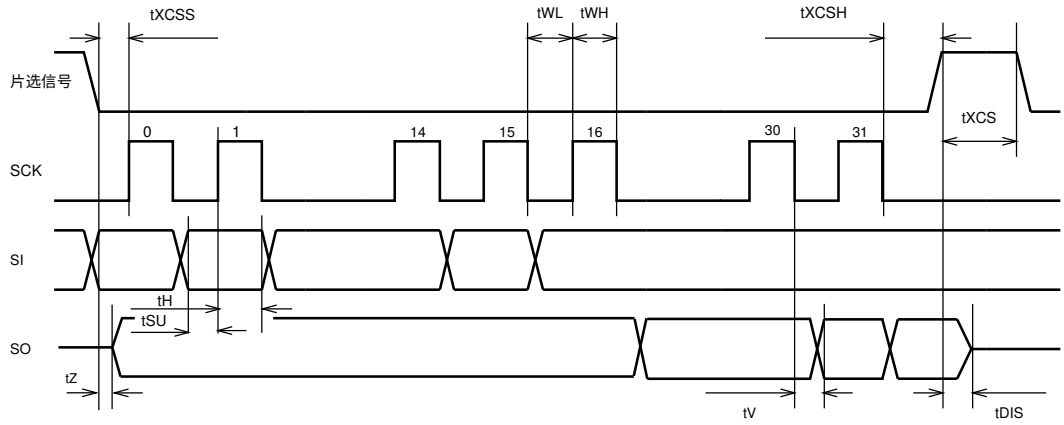


图12：SPI时序图

SCI时序图如图12所示。

符号	最小值	最大值	单位
tXCSS	5		ns
tSU	0		ns
tH	2		特定术语：CLKI 时钟周期
tZ	0		ns
tWL	2		特定术语：CLKI 时钟周期
tWH	2		特定术语：CLKI 时钟周期
tV	2 (+25纳秒[1])		特定术语：CLKI 时钟周期
tXCSH	1		特定术语：CLKI 时钟周期
tXCS	2		特定术语：CLKI 时钟周期
tDIS		10	ns

[1] 引脚负载电容为100 pF时，该时间为25 ns。 电容较低时，该时间更短。

注：虽然时序源自内部时钟CLKI，但系统始终以$1.0\times$模式启动，此时CLKI=XTALI。通过SCI_CLOCKF寄存器配置更高时钟频率并等待DREQ信号上升后，即可使用更高SPI传输速率

注：由于 tWL + tWH + tH = $6\times$CLKI + 25 ns，SCI读取操作的最大速率为 CLKI/7。

7.5 SPI Examples with SM_SDINEW and SM_SDISHARED set

7.5.1 Two SCI Writes

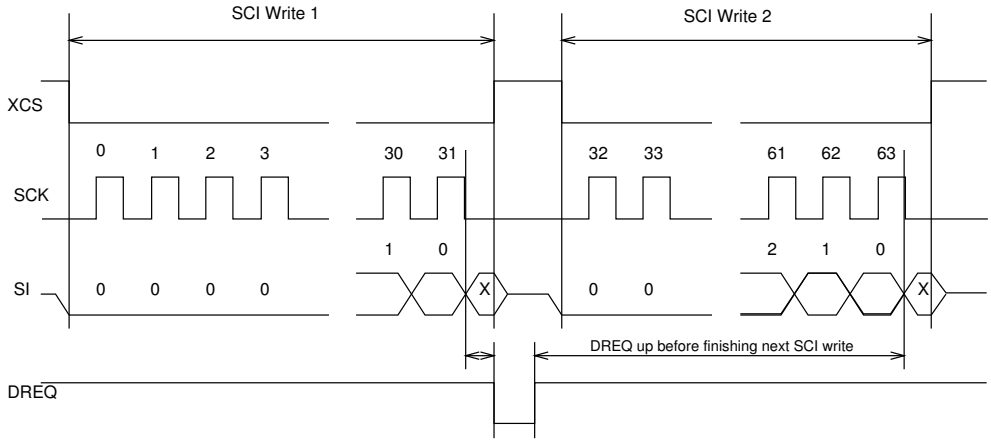


Figure 13: Two SCI operations

Figure 13 shows two consecutive SCI operations. Note that xCS *must* be raised to inactive state between the writes. Also DREQ must be respected as shown in the figure.

7.5.2 Two SDI Bytes

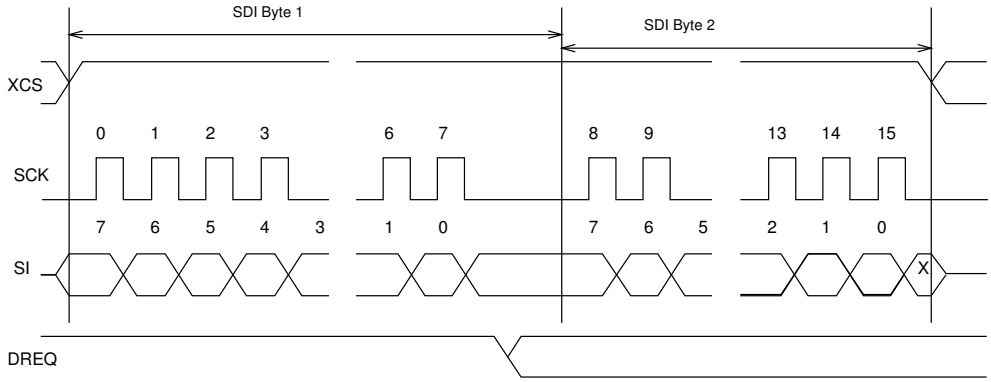


Figure 14: Two SDI bytes

SDI data is synchronized with a raising edge of xCS as shown in Figure 14. However, every byte doesn't need separate synchronization.

7.5 设置 SM_SDINEW 和 SM_SDISHARED 时的 SPI 示例

7.5.1 两次SCI写入操作

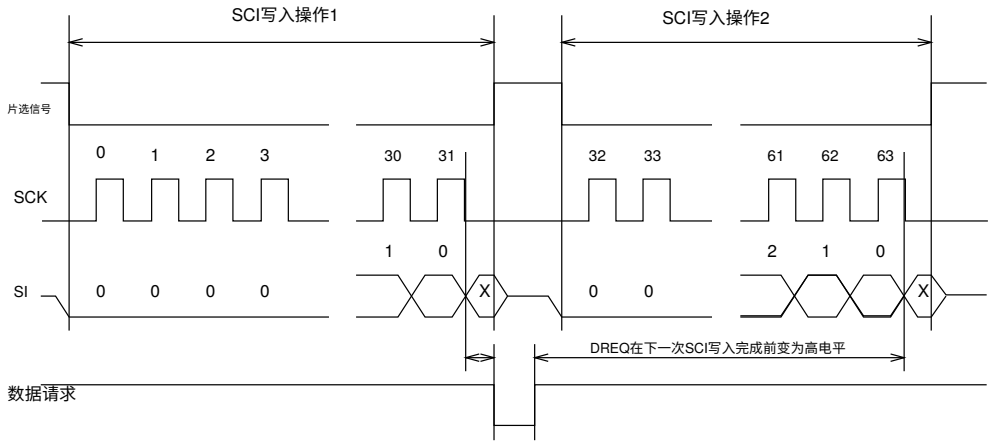


图13：两次SCI操作

图13展示了两次连续的SCI操作。需注意：两次写入操作之间必须将xCS置为无效状态。同时必须遵循图中所示的DREQ信号要求。

7.5.2 两个SDI字节传输

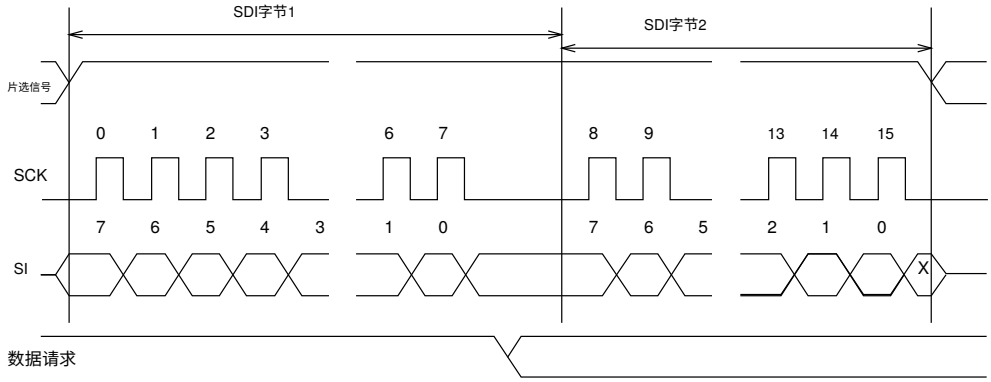


图14：两个SDI字节传输

如图14所示，SDI数据在xCS上升沿同步采样。然而，并非每个字节都需要单独的同步操作。

7.5.3 SCI Operation in Middle of Two SDI Bytes

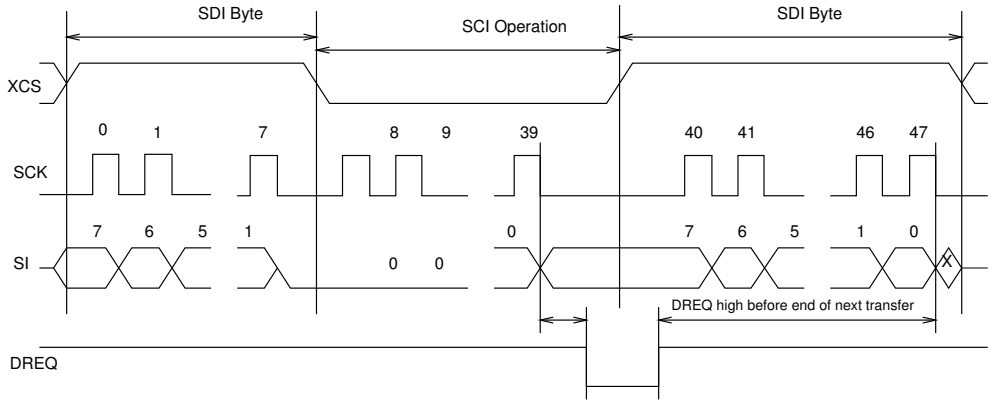


Figure 15: Two SDI bytes separated by an SCI operation

Figure 15 shows how an SCI operation is embedded in between SDI operations. xCS edges are used to synchronize both SDI and SCI. Remember to respect DREQ as shown in the figure.

7.5.3 两个SDI字节之间的SCI操作

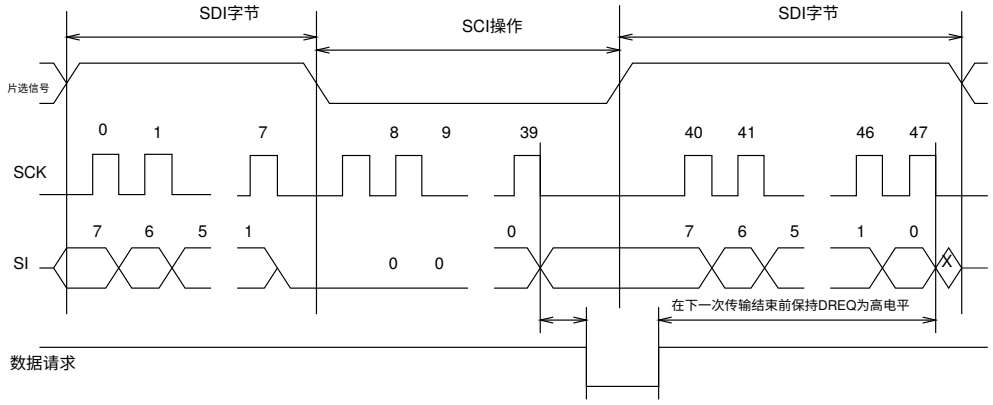


图15：被SCI操作分隔的两个SDI字节

图15展示了SCI操作如何嵌入在SDI操作之间。xCS边沿用于同步SDI和SCI操作。如图所示，请确保遵循DREQ信号要求。

8 Supported Audio Decoder Formats

Conventions	
Mark	**Description**
+	Format is supported
?	Format is supported but not thoroughly tested
-	Format exists but is not supported
	Format doesn't exist

8.1 Supported MP3 (MPEG layer III) Formats

MPEG 1.0[1]:

Samplerate / Hz	Bitrate / kbit/s													
	32	40	48	56	64	80	96	112	128	160	192	224	256	320
48000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
44100	+	+	+	+	+	+	+	+	+	+	+	+	+	+
32000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

MPEG 2.0[1]:

Samplerate / Hz	Bitrate / kbit/s													
	8	16	24	32	40	48	56	64	80	96	112	128	144	160
24000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
22050	+	+	+	+	+	+	+	+	+	+	+	+	+	+
16000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

MPEG 2.5[1]:

Samplerate / Hz	Bitrate / kbit/s													
	8	16	24	32	40	48	56	64	80	96	112	128	144	160
12000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
11025	+	+	+	+	+	+	+	+	+	+	+	+	+	+
8000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

[1] Also all variable bitrate (VBR) formats are supported.

8 支持的音频解码格式

格式标识规范	
标记	**描述**
+	支持该格式
?	支持该格式但未经全面测试
-	格式存在但不支持
	格式不存在

8.1 支持的MP3（MPEG layer III）格式

MPEG 1.0[1]:

采样率（赫兹）	比特率 / 千比特每秒													
	32	40	48	56	64	80	96	112	128	160	192	224	256	320
48000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
44100	+	+	+	+	+	+	+	+	+	+	+	+	+	+
32000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

MPEG 2.0[1]:

采样率（赫兹）	比特率 / 千比特每秒													
	8	16	24	32	40	48	56	64	80	96	112	128	144	160
24000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
22050	+	+	+	+	+	+	+	+	+	+	+	+	+	+
16000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

MPEG 2.5[1]:

采样率（赫兹）	比特率 / 千比特每秒													
	8	16	24	32	40	48	56	64	80	96	112	128	144	160
12000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
11025	+	+	+	+	+	+	+	+	+	+	+	+	+	+
8000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

[1] 同时支持所有可变比特率（VBR）格式。

8.2 Supported MP2 (MPEG layer II) Formats

Note: Layer I / II decoding must be specifically enabled from register SCI_MODE.

MPEG 1.0:

Samplerate / Hz	Bitrate / kbit/s													
	32	48	56	64	80	96	112	128	160	192	224	256	320	384
48000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
44100	+	+	+	+	+	+	+	+	+	+	+	+	+	+
32000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

MPEG 2.0:

Samplerate / Hz	Bitrate / kbit/s													
	8	16	24	32	40	48	56	64	80	96	112	128	144	160
24000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
22050	+	+	+	+	+	+	+	+	+	+	+	+	+	+
16000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

8.3 Supported MP1 (MPEG layer I) Formats

Note: Layer I / II decoding must be specifically enabled from register SCI_MODE.

MPEG 1.0:

Samplerate / Hz	Bitrate / kbit/s													
	32	64	96	128	160	192	224	256	288	320	352	384	416	448
48000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
44100	+	+	+	+	+	+	+	+	+	+	+	+	+	+
32000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

MPEG 2.0:

Samplerate / Hz	Bitrate / kbit/s													
	32	48	56	64	80	96	112	128	144	160	176	192	224	256
24000	?	?	?	?	?	?	?	?	?	?	?	?	?	?
22050	?	?	?	?	?	?	?	?	?	?	?	?	?	?
16000	?	?	?	?	?	?	?	?	?	?	?	?	?	?

8.4 Supported Ogg Vorbis Formats

Parameter	Min	Max	Unit
Channels		2	
Window size	64	4096	samples
Samplerate		48000	Hz
Bitrate		500	kbit/sec

Only floor 1 is supported. No known current encoder uses floor 0. All one- and two-channel Ogg Vorbis files should be playable with this decoder.

8.2 支持的MP2（MPEG第II层）格式

注意：层I/II解码必须通过SCI_MODE寄存器显式启用。

MPEG 1.0:

采样率（赫兹）	比特率 / 千比特每秒													
	32	48	56	64	80	96	112	128	160	192	224	256	320	384
48000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
44100	+	+	+	+	+	+	+	+	+	+	+	+	+	+
32000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

MPEG 2.0:

采样率（赫兹）	比特率 / 千比特每秒													
	8	16	24	32	40	48	56	64	80	96	112	128	144	160
24000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
22050	+	+	+	+	+	+	+	+	+	+	+	+	+	+
16000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

8.3 支持的MP1（MPEG第I层）格式

注意：层I/II解码必须通过SCI_MODE寄存器显式启用。

MPEG 1.0:

采样率（赫兹）	比特率 / 千比特每秒													
	32	64	96	128	160	192	224	256	288	320	352	384	416	448
48000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
44100	+	+	+	+	+	+	+	+	+	+	+	+	+	+
32000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

MPEG 2.0:

采样率（赫兹）	比特率 / 千比特每秒													
	32	48	56	64	80	96	112	128	144	160	176	192	224	256
24000	?	?	?	?	?	?	?	?	?	?	?	?	?	?
22050	?	?	?	?	?	?	?	?	?	?	?	?	?	?
16000	?	?	?	?	?	?	?	?	?	?	?	?	?	?

8.4 支持的Ogg Vorbis格式

参数	最小值	最大值	单位
声道数		2	
窗口大小	64	4096	采样点
采样率		48000	赫兹
比特率		500	千比特/秒

仅支持层级1。 目前已知的编码器均未使用类型0基底。所有单声道和双声道的 Ogg Vorbis 格式文件应均可通过本解码器播放。

8.5 Supported AAC (ISO/IEC 13818-7 and ISO/IEC 14496-3) Formats

VS1053b decodes MPEG2-AAC-LC-2.0.0.0 and MPEG4-AAC-LC-2.0.0.0 streams, i.e. the low complexity profile with maximum of two channels can be decoded. If a stream contains more than one element and/or element type, you can select which one to decode from the 16 single-channel, 16 channel-pair, and 16 low-frequency elements. The default is to select the first one that appears in the stream.

Dynamic range control (DRC) is supported and can be controlled by the user to limit or enhance the dynamic range of the material that contains DRC information.

Both Sine window and Kaiser-Bessel-derived window are supported. For MPEG4 pseudo-random noise substitution (PNS) is supported. Short frames (120 and 960 samples) are not supported.

Spectral Band Replication (SBR) level 3, and Parametric Stereo (PS) level 3 are supported (HE-AAC v2). Level 3 means that maximum of 2 channels, samplerates upto and including 48 kHz without and with SBR (with or without PS) are supported. Also, both mixing modes (R_a and R_b), IPD/OPD synthesis and 34 frequency bands resolution are implemented. The downsampled synthesis mode (core coder rates $>$ 24 kHz and $<=$ 48 kHz with SBR) is implemented.

SBR and PS decoding can also be disabled. Also different operating modes can be selected. See `config1` and `sbrAndPsStatus` in section 10.11 : "Extra parameters".

If enabled, the internal clock (CLKI) is automatically increased if AAC decoding needs a higher clock. PS and SBR operation is automatically switched off if the internal clock is too slow for correct decoding. Generally HE-AAC v2 files need 4.5$\times$ clock to decode both SBR and PS content. This is why 3.5$\times$ + 1.0$\times$ clock is the recommended default.

For AAC the streaming ADTS format is recommended. This format allows easy rewind and fast forward because resynchronization is easily possible.

In addition to ADTS (.aac), MPEG2 ADIF (.aac) and MPEG4 AUDIO (.mp4 / .m4a) files are played, but these formats are less suitable for rewind and fast forward operations. You can still implement these features by using the safe jump points table, or using slightly less robust but much easier automatic resync mechanism (see Section 10.5.4).

Because 3GPP (.3gp) and 3GPPv2 (.3g2) files are just MPEG4 files, those that contain only HE-AAC or HE-AACv2 content are played.

Note: To be able to play the .3gp, .3g2, .mp4 and .m4a files, the **mdat** atom must be the last atom in the MP4 file. Because VS1053b receives all data as a stream, all metadata must be available before the music data is received. Several MP4 file formatters do not satisfy this requirement and some kind of conversion is required. This is also why the streamable ADTS format is recommended.

Programs exist that optimize the .mp4 and .m4a into so-called *streamable* format that has the

8.5 支持的 AAC 格式（ISO/IEC 13818-7 与 ISO/IEC 14496-3）

VS1053b 可解码 MPEG2-AAC-LC-2.0.0.0 和 MPEG4-AAC-LC-2.0.0.0 流，即支持解码低复杂度配置文件、最多包含两个声道的音频流。若数据流包含多个元素和/或元素类型，用户可从16个单声道元素、16个声道对元素及16个低频元素中选择需解码的元素。默认选择数据流中出现的首个元素。

支持动态范围控制（DRC），用户可通过该功能对包含DRC信息的素材进行动态范围限制或增强。

支持正弦窗和凯塞-贝塞尔派生窗。支持MPEG4伪随机噪声替换(PNS)功能。不支持短帧（120及960个采样点）。

支持谱带复制(SBR)级别3与参数立体声(PS)级别3（HE-AAC v2标准）。 级别3表示支持最多2声道、采样率最高达48 kHz（含）的配置，无论是否启用SBR（含PS启用状态）。 同时实现了混合模式（R_a与 R_b）、强度/声像差(IPD/OPD)合成及34个频带解析功能。实现下采样合成模式（核心编码器速率 $>$24 kHz且 $<=$48 kHz并启用SBR时）。

可禁用SBR与PS解码功能。可选择不同操作模式。
详见第10.11节"扩展参数"中的 config1与sbrAndPsStatus寄存器说明。

若启用此功能，当AAC解码需要更高时钟频率时，内部时钟(CLKI)将自动提升。若内部时钟过慢而无法正确解码，PS与SBR功能将自动关闭。通常HE-AAC v2文件需要4.5×时钟频率方可同时解码SBR和PS内容。因此推荐将3.5×+1.0×时钟设为默认配置。

对于AAC格式，建议采用流式ADTS封装。该格式支持便捷的回退与快进操作，因其具备高效的重同步能力。

除ADTS(.aac)外，亦支持播放MPEG2 ADIF(.aac)及MPEG4 AUDIO(.mp4/.m4a)文件，但后两种格式的回退与快进操作性较差。仍可通过安全跳点表实现这些功能，或采用稍欠稳健但更简易的自动重同步机制（参阅章节10.5.4）。

由于3GPP(.3gp)和3GPPv2(.3g2)文件本质上是MPEG4文件，因此仅包含HE-AAC或HE-AACv2编码内容的文件可被播放。

注意：要支持播放.3gp、.3g2、.mp4和.m4a文件，其MP4文件中的 **mdat**原子必须是最后一个原子结构。由于VS1053b以流形式接收所有数据，所有元数据必须在接收音乐数据前完整就绪。许多MP4文件格式化工具无法满足此要求，因此需要进行某种格式转换。这也是推荐采用可流式传输的ADTS格式的原因。

现有程序可将.mp4和.m4a文件优化为所谓的流式格式，

mdat atom last in the file, and thus suitable for web servers' audio streaming. You can use this kind of tool to process files for VS1053b too. For example `mp4creator -optimize file.mp4`.

AAC[1][2]:

Samplerate / Hz	Maximum Bitrate kbit/s - for 2 channels								
	≤96	132	144	192	264	288	384	529	576
48000	+	+	+	+	+	+	+	+	+
44100	+	+	+	+	+	+	+	+	
32000	+	+	+	+	+	+	+		
24000	+	+	+	+	+	+			
22050	+	+	+	+	+				
16000	+	+	+	+					
12000	+	+	+						
11025	+	+							
8000	+								

[1] 64000 Hz, 88200 Hz, and 96000 Hz AAC files are played at the highest possible samplerate (48000 Hz with 12.288 MHz XTALI).

[2] Also all variable bitrate (VBR) formats are supported. Note that the table gives the maximum bitrate allowed for two channels for a specific samplerate as defined by the AAC specification. The decoder does not actually have a fixed lower or upper limit.

使文件末尾存放mdat原子结构，从而适配网络服务器的音频流传输需求。您也可使用此类工具处理VS1053b的文件。例如mp4creator -optimize file.mp4。

AAC[12]:

采样率（赫兹）	最大比特率 kbit/s - 双声道模式								
	≤96	132	144	192	264	288	384	529	576
48000	+	+	+	+	+	+	+	+	+
44100	+	+	+	+	+	+	+	+	
32000	+	+	+	+	+	+	+		
24000	+	+	+	+	+	+			
22050	+	+	+	+	+				
16000	+	+	+	+					
12000	+	+	+						
11025	+	+							
8000	+								

[1] 64000赫兹、88200赫兹及96000赫兹的AAC文件将以最高可能采样率播放(使用12.288 MHz晶振时为48000赫兹)。

[2] 同时支持所有可变比特率(VBR)格式。请注意：表格所示是根据AAC规范定义的特定采样率下双声道允许的最大比特率。
解码器实际并无固定的下限或上限限制。

8.6 Supported WMA Formats

Windows Media Audio codec versions 2, 7, 8, and 9 are supported. All WMA profiles (L1, L2, and L3) are supported. Previously streams were separated into Classes 1, 2a, 2b, and 3. The decoder has passed Microsoft's conformance testing program. Windows Media Audio Professional is a different codec and is not supported.

WMA 4.0 / 4.1:

Samplerate	**Bitrate / kbit/s**																
/ Hz	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192
8000	+	+	+		+												
11025			+	+													
16000				+	+	+	+										
22050						+	+	+	+								
32000							+	+	+	+	+	+					
44100									+		+	+	+	+	+	+	
48000															+	+	

WMA 7:

Samplerate	**Bitrate / kbit/s**																
/ Hz	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192
8000	+	+	+		+												
11025			+	+													
16000				+	+	+	+										
22050						+	+	+	+								
32000							+		+	+	+						
44100									+		+	+	+	+	+	+	+
48000															+	+	

WMA 8:

Samplerate	**Bitrate / kbit/s**																
/ Hz	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192
8000	+	+	+		+												
11025			+	+													
16000				+	+	+	+										
22050						+	+	+	+								
32000							+		+	+	+						
44100									+		+	+	+	+	+	+	+
48000															+	+	+

WMA 9:

Samplerate	**Bitrate / kbit/s**																		
/ Hz	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192	256	320
8000	+	+	+		+														
11025			+	+															
16000				+	+	+	+												
22050						+	+	+	+										
32000							+		+	+	+								
44100							+		+		+	+	+	+	+	+	+	+	+
48000												+		+	+	+	+		

In addition to these expected WMA decoding profiles, all other bitrate and samplerate combinations are supported, including variable bitrate WMA streams. Note that WMA does not consume the bitstream as evenly as MP3, so you need a higher peak transfer capability for clean playback at the same bitrate.

8.6 支持的WMA格式

支持Windows Media Audio编解码器版本2、7、8及9。 支持所有WMA配置文件(L1、L2和L3)。 先前流媒体被划分为1、2a、2b和3等级。该解码器已通过微软符合性测试项目认证。Windows Media Audio Professional是独立编解码器，且不受支持。

WMA 4.0 / 4.1:

采样率	比特率 / 千比特每秒																
/ 赫兹	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192
8000	+	+	+		+												
11025			+	+													
16000				+	+	+	+										
22050						+	+	+	+								
32000							+	+	+	+	+	+					
44100									+		+	+	+	+	+	+	
48000															+	+	

WMA 7:

采样率	比特率 / 千比特每秒																
/ 赫兹	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192
8000	+	+	+		+												
11025			+	+													
16000				+	+	+	+										
22050						+	+	+	+								
32000							+		+	+	+						
44100									+		+	+	+	+	+	+	+
48000															+	+	

WMA 8:

采样率	比特率 / 千比特每秒																
/ 赫兹	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192
8000	+	+	+		+												
11025			+	+													
16000				+	+	+	+										
22050						+	+	+	+								
32000							+		+	+	+						
44100									+		+	+	+	+	+	+	+
48000															+	+	+

WMA 9:

采样率	比特率 / 千比特每秒																		
/ 赫兹	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192	256	320
8000	+	+	+		+														
11025			+	+															
16000				+	+	+	+												
22050						+	+	+	+										
32000							+		+	+	+								
44100							+		+		+	+	+	+	+	+	+	+	+
48000												+		+	+	+	+		

除上述常规WMA解码方案外，所有其他比特率与采样率组合均受支持，包括可变比特率WMA流。需注意WMA处理比特流的均匀性不及MP3，因此在相同比特率下需要更高的峰值传输能力才能实现无杂音播放。

8.7 Supported FLAC Formats

Upto 48 kHz and 24-bit FLAC files are supported with the *VS1053b Patches w/ FLAC Decoder* plugin that is available at *http://www.vlsi.fi/en/support/software/vs10xxplugins.html* . Read the accompanying documentation of the plugin for details.

8.8 Supported RIFF WAV Formats

The most common RIFF WAV subformats are supported, with 1 or 2 audio channels.

Format	Name	Supported	Comments
0x01	PCM	+	16 and 8 bits, any samplerate ≤ 48kHz
0x02	ADPCM	-	
0x03	IEEE_FLOAT	-	
0x06	ALAW	-	
0x07	MULAW	-	
0x10	OKI_ADPCM	-	
0x11	IMA_ADPCM	+	Any samplerate ≤ 48kHz
0x15	DIGISTD	-	
0x16	DIGIFIX	-	
0x30	DOLBY_AC2	-	
0x31	GSM610	-	
0x3b	ROCKWELL_ADPCM	-	
0x3c	ROCKWELL_DIGITALK	-	
0x40	G721_ADPCM	-	
0x41	G728_CELP	-	
0x50	MPEG	-	
0x55	MPEGLAYER3	+	For supported MP3 modes, see Chapter 8.1
0x64	G726_ADPCM	-	
0x65	G722_ADPCM	-	

8.7 支持的FLAC格式

*VS1053*b通过安装FLAC解码器插件可支持最高48 kHz采样率及24位深度的FLAC文件，该插件可在*http://www.*vlsi.fi*/en/support/software/vs10*xxplugin*s.html*获取。详情请参阅该插件的随附文档。

8.8 支持的RIFF WAV格式

支持最常见的RIFF WAV子格式，包含1或2个音频声道。

格式	名称	是否支持	注释
0x01	PCM	+	16位与8位，任意采样率 $\leq$ 48kHz
0x02	特定术语：ADPCM	-	
0x03	IEEE_FLOAT	-	
0x06	ALAW	-	
0x07	MULAW	-	
0x10	OKI_ADPCM	-	
0x11	IMA_ADPCM	+	任意采样率 $\leq$ 48kHz
0x15	DIGISTD	-	
0x16	DIGIFIX	-	
0x30	DOLBY_AC2	-	
0x31	GSM610	-	
0x3b	ROCKWELL_ADPCM	-	
0x3c	ROCKWELL_DIGITALK	-	
特定术语：0x40	G721_ADPCM	-	
0x41	G728_CELP	-	
0x50	MPEG	-	
0x55	MPEGLAYER3	+	支持的MP3模式详见第8.1章
0x64	G726_ADPCM	-	
0x65	G722_ADPCM	-	

8.9 Supported MIDI Formats

General MIDI and SP-MIDI format 0 files are played. Format 1 and 2 files must be converted to format 0 by the user. The maximum polyphony is 64, the maximum sustained polyphony is 40. Actual polyphony depends on the internal clock rate (which is user-selectable), the instruments used, whether the reverb effect is enabled, and the possible global postprocessing effects enabled, such as bass enhancer, treble control or EarSpeaker spatial processing. The polyphony restriction algorithm makes use of the SP-MIDI MIP table, if present, and uses smooth note removal.

43 MHz ($3.5\times$ input clock) achieves 19-31 simultaneous sustained notes. The instantaneous amount of notes can be larger. This is a fair compromise between power consumption and quality, but higher clocks can be used to increase polyphony.

Reverb effect can be controlled by the user. In addition to reverb automatic and reverb off modes, 14 different decay times can be selected. These roughly correspond to different room sizes. Also, each midi song decides how much effect each instrument gets. Because the reverb effect uses about 4 MHz of processing power the automatic control enables reverb only when the internal clock is at least $3.0\times$.

In VS1053b both EarSpeaker and MIDI reverb can be on simultaneously. This is ideal for listening MIDI songs with headphones.

New instruments have been implemented in addition to the 36 that are available in VS1003. VS1053b now has unique instruments in the whole GM1 instrument set and one bank of GM2 percussions.

Supported MIDI messages:

- meta: 0x51 : set tempo
- other meta: MidiMeta() called
- device control: 0x01 : master volume
- channel message: 0x80 note off, 0x90 note on, 0xc0 program, 0xe0 pitch wheel
- channel message 0xb0: parameter
  - 0x00: bank select (0 is default, 0x78 and 0x7f is drums, 0x79 melodic)
  - 0x06: RPN MSB: 0 = bend range, 2 = coarse tune
  - 0x07: channel volume
  - 0x0a: pan control
  - 0x0b: expression (changes volume)
  - 0x0c: effect control 1 (sets global reverb decay)
  - 0x26: RPN LSB: 0 = bend range
  - 0x40: hold1
  - 0x42: sustenuto
  - 0x5b effects level (channel reverb level)
  - 0x62,0x63,0x64,0x65: NRPN and RPN selects
  - 0x78: all sound off
  - 0x79: reset all controllers
  - 0x7b, 0x7c, 0x7d: all notes off

8.9 支持的MIDI格式

可播放通用MIDI及SP-MIDI格式0文件。格式1和格式2文件必须由用户转换为格式0。 最大复音数为64，最大持续复音数为40。
实际复音数取决于内部时钟速率（用户可配置）、使用的乐器、是否启用混响效果，以及可能启用的全局后处理效果（如低音增强器、高音控制或耳机/扬声器空间处理）。 复音数限制算法会调用SP-MIDI MIP表（若存在），并采用平滑音符消除机制。

43 MHz（$3.5\times$输入时钟）可实现19-31个同步持续音符。瞬时音符数量可能更高。这是功耗与音质间的合理折衷方案，但可通过提升时钟频率来增加复音数。

用户可控制混响效果。除自动混响和关闭混响模式外，还可选择14种不同的衰减时间。这些参数大致对应不同房间尺寸的声学特性。此外，每首MIDI乐曲自行决定各乐器所获效果强度。因混响效果需消耗约4兆赫兹处理能力，自动控制仅在内部时钟频率$\geqslant 3.0\times$时启用该效果。

VS1053b可同时启用耳机/扬声器与MIDI混响功能。此特性特别适合通过耳机欣赏MIDI乐曲。

在VS1003原有36种乐器基础上，新增实现了若干新乐器。
VS1053b现拥有GM1乐器组中独特音色及全套GM2打击乐器库。

支持的MIDI消息类型：

- 元事件：0x51 : 设置速度
- 其他元事件：调用MidiMeta()函数
- 设备控制：0x01 : 主音量
- 通道消息：0x80 音符关闭, 0x90 音符开启, 0xc0 音色切换, 0xe0 弯音轮
- 通道消息 0xb0：参数
  - 0x00：音色库选择（默认为0，0x78与0x7f为鼓组音色，0x79为旋律音色）
  - 0x06：RPN最高有效位：0=弯音范围，2=粗调音高
  - 0x07：通道音量
  - 0x0a：声像控制
  - 0x0b：表情控制器（调节音量）
  - 0x0c：效果控制1（设置全局混响衰减）
  - 0x26：RPN最低有效位：0=弯音范围
  - 0x40：延音踏板1
  - 0x42：持续音踏板
  - 0x5b：效果电平（通道混响电平）
  - 0x62,0x63,0x64,0x65：NRPN与RPN选择器
  - 0x78：关闭所有声音
  - 0x79：复位所有控制器
  - 0x7b, 0x7c, 0x7d：关闭所有音符

VS1053b Melodic Instruments (GM1)			
1 Acoustic Grand Piano	33 Acoustic Bass	65 Soprano Sax	97 Rain (FX 1)
2 Bright Acoustic Piano	34 Electric Bass (finger)	66 Alto Sax	98 Sound Track (FX 2)
3 Electric Grand Piano	35 Electric Bass (pick)	67 Tenor Sax	99 Crystal (FX 3)
4 Honky-tonk Piano	36 Fretless Bass	68 Baritone Sax	100 Atmosphere (FX 4)
5 Electric Piano 1	37 Slap Bass 1	69 Oboe	101 Brightness (FX 5)
6 Electric Piano 2	38 Slap Bass 2	70 English Horn	102 Goblins (FX 6)
7 Harpsichord	39 Synth Bass 1	71 Bassoon	103 Echoes (FX 7)
8 Clavi	40 Synth Bass 2	72 Clarinet	104 Sci-fi (FX 8)
9 Celesta	41 Violin	73 Piccolo	105 Sitar
10 Glockenspiel	42 Viola	74 Flute	106 Banjo
11 Music Box	43 Cello	75 Recorder	107 Shamisen
12 Vibraphone	44 Contrabass	76 Pan Flute	108 Koto
13 Marimba	45 Tremolo Strings	77 Blown Bottle	109 Kalimba
14 Xylophone	46 Pizzicato Strings	78 Shakuhachi	110 Bag Pipe
15 Tubular Bells	47 Orchestral Harp	79 Whistle	111 Fiddle
16 Dulcimer	48 Timpani	80 Ocarina	112 Shanai
17 Drawbar Organ	49 String Ensembles 1	81 Square Lead (Lead 1)	113 Tinkle Bell
18 Percussive Organ	50 String Ensembles 2	82 Saw Lead (Lead)	114 Agogo
19 Rock Organ	51 Synth Strings 1	83 Calliope Lead (Lead 3)	115 Pitched Percussion
20 Church Organ	52 Synth Strings 2	84 Chiff Lead (Lead 4)	116 Woodblock
21 Reed Organ	53 Choir Aahs	85 Charang Lead (Lead 5)	117 Taiko Drum
22 Accordion	54 Voice Oohs	86 Voice Lead (Lead 6)	118 Melodic Tom
23 Harmonica	55 Synth Voice	87 Fifths Lead (Lead 7)	119 Synth Drum
24 Tango Accordion	56 Orchestra Hit	88 Bass + Lead (Lead 8)	120 Reverse Cymbal
25 Acoustic Guitar (nylon)	57 Trumpet	89 New Age (Pad 1)	121 Guitar Fret Noise
26 Acoustic Guitar (steel)	58 Trombone	90 Warm Pad (Pad 2)	122 Breath Noise
27 Electric Guitar (jazz)	59 Tuba	91 Polysynth (Pad 3)	123 Seashore
28 Electric Guitar (clean)	60 Muted Trumpet	92 Choir (Pad 4)	124 Bird Tweet
29 Electric Guitar (muted)	61 French Horn	93 Bowed (Pad 5)	125 Telephone Ring
30 Overdriven Guitar	62 Brass Section	94 Metallic (Pad 6)	126 Helicopter
31 Distortion Guitar	63 Synth Brass 1	95 Halo (Pad 7)	127 Applause
32 Guitar Harmonics	64 Synth Brass 2	96 Sweep (Pad 8)	128 Gunshot

VS1053b Percussion Instruments (GM1+GM2)			
27 High Q	43 High Floor Tom	59 Ride Cymbal 2	75 Claves
28 Slap	44 Pedal Hi-hat [EXC 1]	60 High Bongo	76 Hi Wood Block
29 Scratch Push [EXC 7]	45 Low Tom	61 Low Bongo	77 Low Wood Block
30 Scratch Pull [EXC 7]	46 Open Hi-hat [EXC 1]	62 Mute Hi Conga	78 Mute Cuica [EXC 4]
31 Sticks	47 Low-Mid Tom	63 Open Hi Conga	79 Open Cuica [EXC 4]
32 Square Click	48 High Mid Tom	64 Low Conga	80 Mute Triangle [EXC 5]
33 Metronome Click	49 Crash Cymbal 1	65 High Timbale	81 Open Triangle [EXC 5]
34 Metronome Bell	50 High Tom	66 Low Timbale	82 Shaker
35 Acoustic Bass Drum	51 Ride Cymbal 1	67 High Agogo	83 Jingle bell
36 Bass Drum 1	52 Chinese Cymbal	68 Low Agogo	84 Bell tree
37 Side Stick	53 Ride Bell	69 Cabasa	85 Castanets
38 Acoustic Snare	54 Tambourine	70 Maracas	86 Mute Surdo [EXC 6]
39 Hand Clap	55 Splash Cymbal	71 Short Whistle [EXC 2]	87 Open Surdo [EXC 6]
40 Electric Snare	56 Cowbell	72 Long Whistle [EXC 2]	
41 Low Floor Tom	57 Crash Cymbal 2	73 Short Guiro [EXC 3]	
42 Closed Hi-hat [EXC 1]	58 Vibra-slap	74 Long Guiro [EXC 3]	

VS1053b 旋律乐器 (GM1)			
1 大钢琴	33 原声贝司	65 高音萨克斯	97 雨声 (效果1)
2 亮音钢琴	34 指拨电贝司	66 中音萨克斯	98 音轨 (效果2)
3 电钢琴	35 拨片电贝司	67 次中音萨克斯	99 水晶声 (效果3)
4 酒吧钢琴	36 无品贝司	68 上低音萨克斯	100 大气氛围 (效果4)
5 电钢琴1	37 击弦贝斯 1	69 双簧管	101 明亮度 (FX 5)
6 电钢琴 2	38 击弦贝斯 2	70 英国管	102 哥布林音效 (FX 6)
7 大键琴	39 合成贝斯 1	73 巴松管	103 回声效果 (FX 7)
8 击弦古钢琴	40 合成贝斯 2	72 单簧管	104 科幻音效 (FX 8)
9 钢片琴	41 小提琴	73 短笛	105 西塔尔琴
10 钟琴	42 中提琴	74 长笛	106 班卓琴
11 八音盒	43 大提琴	75 直笛	107 三味线
12 颤音琴	44 倍低音提琴	76 排箫	108 筝
13 马林巴琴	45 颤音弦乐	77 吹瓶	109 卡林巴琴
14 木琴	46 拨奏弦乐	78 尺八	110 风笛
15 管钟	47 管弦竖琴	79 哨笛	111 民间提琴
16 扬琴	48 定音鼓	80 陶笛	112 唢呐
17 拉杆风琴	49 弦乐合奏1	81 方波主奏 (主奏1)	113 叮当铃
18 击打风琴	50 弦乐合奏2	82 锯齿波主奏 (主奏)	114 阿哥哥
19 摇滚风琴	51 合成弦乐1	83 汽笛风琴主奏 (主奏3)	115 音高打击乐
20 教堂管风琴	52 合成弦乐2	84 雪弗主奏 (主奏4)	116 木鱼
21 簧风琴	53 人声合唱啊	85 夏朗主奏 (主奏5)	117 太鼓
22 手风琴	54 人声合唱哦	86 人声领奏 (主奏6)	118 嗵鼓
23 口琴	55 合成人声	87 五度主奏音色 (主奏7)	119 合成鼓
24 探戈手风琴	56 管弦乐强音	88 贝斯+主奏 (主奏8)	120 逆钹
25 尼龙弦原声吉他	57 小号	89 新世纪音色 (铺垫1)	121 吉他品丝噪声
26 钢弦原声吉他	58 长号	90 温暖铺垫音 (铺垫2)	122 气息噪声
27 爵士电吉他	59 大号	91 复音合成音 (铺垫3)	123 海浪声
28 电吉他(清音)	60 弱音小号	92 合唱音色(背景4)	124 鸟鸣声
29 电吉他(闷音)	61 圆号	93 弓弦音色(背景5)	125 电话铃声
30 过载吉他	62 铜管乐组	94 金属音色(背景6)	126 直升机声
31 失真吉他	63 合成铜管乐1	95 光晕音色(背景7)	127 掌声
32 吉他泛音	64 合成铜管音色 2	96 扫频音效 (合成背景音 8)	128 枪击音效

VS1053b 打击乐器 (GM1+GM2标准)			
27 高频滴答声	43 高音地嗵鼓	59 骑钹 2	75 击棒
28 拍击音效	44 踏板踩镲 [特殊 1]	60 高音邦戈鼓	76 高音梆子
29 搓盘前推 [特殊 7]	45 低音嗵鼓	61 低音邦戈鼓	77 低音梆子
30 搓盘后拉 [特殊 7]	46 开合踩镲 [特殊 1]	62 闷音高康加鼓	78 闷音葵卡鼓 [EXC 4]
31 鼓槌声	47 中低音通鼓	63 开音高康加鼓	79 开音葵卡鼓 [EXC 4]
32 方形咔嗒声	48 高中音通鼓	64 低音康加鼓	80 闷音三角铁 [EXC 5]
33 节拍器咔嗒声	49 碎音钹1	65 高音天巴雷鼓	81 开音三角铁 [EXC 5]
34 节拍器铃声	50 高音通鼓	66 低音天巴雷鼓	82 沙锤
35 原声底鼓	51 叮嚓镲1	67 高音阿哥哥	83 铃铛
36 底鼓1	52 中国镲	68 低音阿哥哥	84 摇铃树
37 边击	53 叮铃镲	69 沙锤	85 响板
38 原声军鼓	54 铃鼓	70 沙槌	86 弱音苏尔朵鼓 [EXC 6]
39 拍手	55 溅镲	71 短哨音 [EXC 2]	87 开合苏尔朵鼓 [EXC 6]
40 电子军鼓	56 牛铃	72 长哨音 [EXC 2]	
41 低音地嗵鼓	57 碎音钹2	73 短刮瓜 [EXC 3]	
42 闭合踩镲 [EXC 1]	58 震音棒	74 长刮瓜 [EXC 3]	

9 Functional Description

9.1 Main Features

VS1053b is based on a proprietary digital signal processor, VS_DSP. It contains all the code and data memory needed for Ogg Vorbis, MP3, AAC, WMA and WAV PCM + ADPCM audio decoding and a MIDI synthesizer, together with serial interfaces, a multirate stereo audio DAC and analog output amplifiers and filters. Also PCM/ADPCM audio encoding is supported using a microphone amplifier and/or line-level inputs and a stereo A/D converter. With software plugins the chip can also decode lossless FLAC as well as record the high-quality Ogg Vorbis format. A UART is provided for debugging purposes.

9 功能描述

9.1 主要特性

VS1053b基于专有数字信号处理器VS_DSP。该芯片集成了支持Ogg Vorbis、MP3、AAC、WMA、WAV PCM + ADPCM音频解码及MIDI合成器所需的全部代码与数据存储器，同时包含串行接口、多速率立体声音频DAC、模拟输出放大器及滤波器模块。此外，通过麦克风放大器和/或线路电平输入接口配合立体声A/D转换器，可支持PCM/ADPCM音频编码功能。通过软件插件，该芯片还能解码无损FLAC格式，并支持录制高品质的Ogg Vorbis格式音频。提供UART接口用于调试目的。

9.2 Data Flow of VS1053b

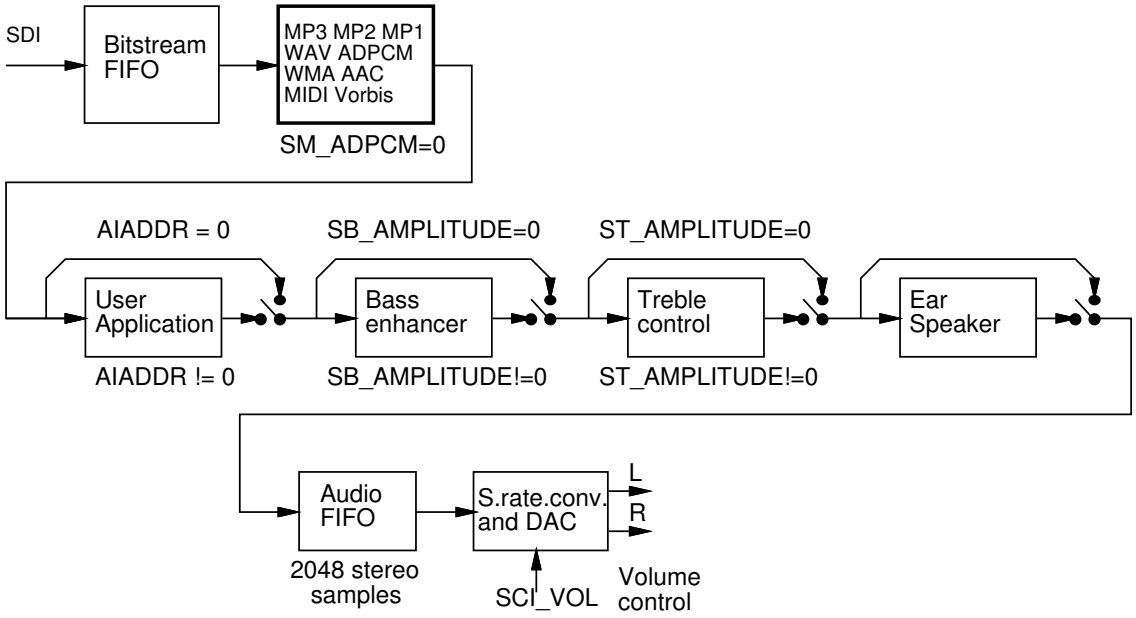


Figure 16: Data flow of VS1053b.

First, depending on the audio data, and provided ADPCM encoding mode is not set, Ogg Vorbis, PCM WAV or IMA ADPCM WAV is received and decoded from the SDI bus.

After decoding, if SCI_AIADDR is non-zero, application code is executed from the address pointed to by that register. For more details, see Application Notes for VS10XX.

Then data may be sent to the Bass Enhancer and Treble Control depending on the SCI_BASS register.

Next, headphone processing is performed, if the EarSpeaker spatial processing is active.

After that the data to the Audio FIFO, which holds the data until it is read by the Audio interrupt and fed to the samplerate converter and DACs. The size of the audio FIFO is 2048 stereo ($2\times$16-bit) samples, or 8 KiB.

The samplerate converter upsamples all different samplerates to XTALI/2, or 128 times the highest usable samplerate with 18-bit precision. Volume control is performed in the upsampled domain. New volume settings are loaded only when the upsampled signal crosses the zero point (or after a timeout). This zero-crossing detection almost completely removes all audible noise that occurs when volume is suddenly changed.

The samplerate conversion to a common samplerate removes the need for complex PLL-based clocking schemes and allows almost unlimited sample rate accuracy with one fixed input clock frequency. With a 12.288 MHz clock, the DA converter operates at 128×48 kHz, i.e. 6.144 MHz, and creates a stereo in-phase analog signal. The oversampled output is low-pass filtered by an on-chip analog filter. This signal is then forwarded to the earphone amplifier.

9.2 VS1053b数据流

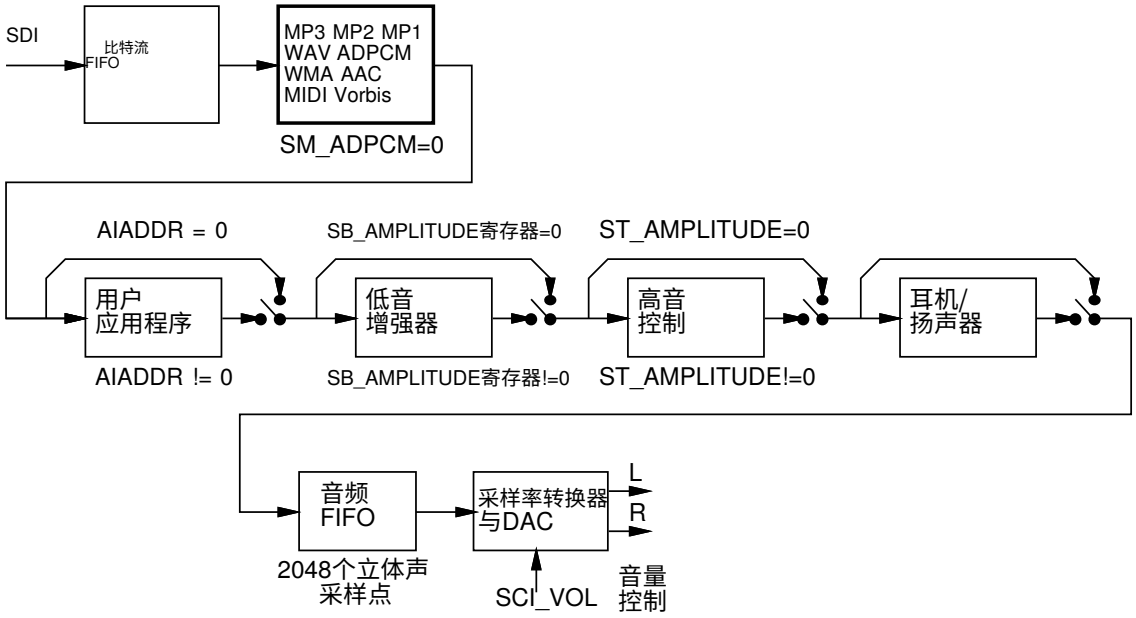


图16：VS1053b 数据流示意图。

首先，根据音频数据类型（若未启用ADPCM编码模式），系统通过SDI总线接收并解码Ogg Vorbis、PCM WAV或IMA ADPCM WAV格式数据。

解码完成后，若SCI_AIADDR寄存器值非零，则执行该寄存器指向地址的应用程序代码。更多细节请参阅《VS10XX系列应用说明》。

随后根据SCI_BASS寄存器配置，数据可能被送往低音增强器和高音控制器处理。

若启用耳机空间处理功能，接下来将执行耳机信号处理。

此后数据进入音频FIFO（容量为2048个立体声样本，即2×16位格式，共8KiB），由音频中断服务程序读取后送至采样率转换器和DAC。

采样率转换器将所有不同采样率上采样至XTALI/2（即最高可用采样率的128倍），并保持18位精度。音量控制在经过上采样的信号域内执行。新的音量设置仅在采样信号经过零点时加载（或超时后强制加载）。 这种过零检测机制几乎完全消除了音量突变时产生的所有可闻噪声。

通过统一采样率的转换机制，无需复杂的基于PLL的时钟方案，在固定输入时钟频率下即可实现近乎无限制的采样率精度。采用12.288 MHz时钟时，DA转换器以 128×48kHz（即6.144 MHz）频率运行，生成立体声同相信号。经片上模拟滤波器对过采样输出进行低通滤波。该信号随后传输至耳机放大器。

9.3 EarSpeaker Spatial Processing

While listening to headphones the sound has a tendency to be localized inside the head. The sound field becomes flat and lacking the sensation of dimensions. This is an unnatural, awkward and sometimes even disturbing situation. This phenomenon is often referred in literature as ‘lateralization’, meaning ’in-the-head’ localization. Long-term listening to lateralized sound may lead to listening fatigue.

All real-life sound sources are external, leaving traces to the acoustic wavefront that arrives to the ear drums. From these traces, the auditory system of the brain is able to judge the distance and angle of each sound source. In loudspeaker listening the sound is external and these traces are available. In headphone listening these traces are missing or ambiguous.

EarSpeaker processes sound to make listening via headphones more like listening to the same music from real loudspeakers or live music. Once EarSpeaker processing is activated, the instruments are moved from inside to the outside of the head, making it easier to separate the different instruments (see figure 17). The listening experience becomes more natural and pleasant, and the stereo image is sharper as the instruments are widely on front of the listener instead of being inside the head.

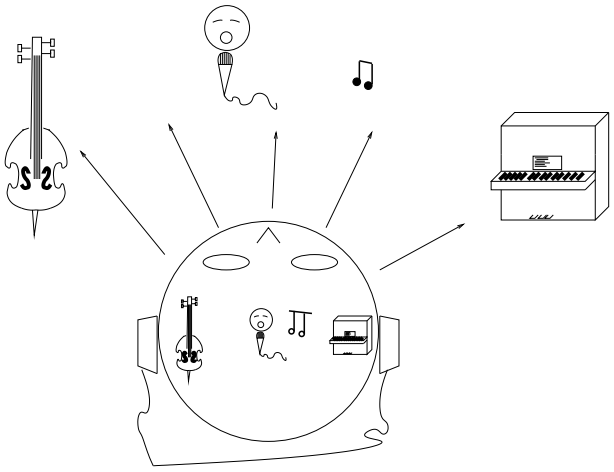


Figure 17: EarSpeaker externalized sound sources vs. normal inside-the-head sound

Note that EarSpeaker differs from any common spatial processing effects, such as echo, reverb, or bass boost. EarSpeaker accurately simulates the human auditory model and real listening environment acoustics. Thus is does not change the tonal character of the music by introducing artificial effects.

EarSpeaker processing can be parameterized to a few different modes, each simulating a little different type of acoustical situation, suiting different personal preferences and types of recording. See section 9.6.1 for how to activate different modes.

- *Off*: Best option when listening through loudspeakers or if the audio to be played contains binaural preprocessing.
- *minimal*: Suited for listening to normal musical scores with headphones, very subtle.

9.3 耳机/扬声器空间处理技术

使用耳机聆听时，声音往往被定位于头部内部。声场会变得扁平化，丧失空间维度感。这种状态不自然、别扭，有时甚至令人不适。该现象在文献中常称为'侧化效应'，即声音在头内的定位现象。长期聆听侧化声音可能导致听觉疲劳。

现实中的所有声源均来自外部，会在抵达耳膜的声波波前留下痕迹。大脑听觉系统能根据这些痕迹判断各声源的距离与角度。扬声器播放时声音源自外部，这些声学痕迹完整存在。耳机聆听时这些痕迹缺失或模糊不清。

耳机/扬声器技术通过声音处理，使耳机聆听体验趋近真实扬声器或现场音乐效果。启用耳机/扬声器处理后，乐器声将从头部内部移至外部，更易区分不同乐器（见图17）。 当乐器声源仿佛广泛分布于聆听者前方而非颅内时，听觉体验将更自然舒适，立体声像亦更为鲜明。

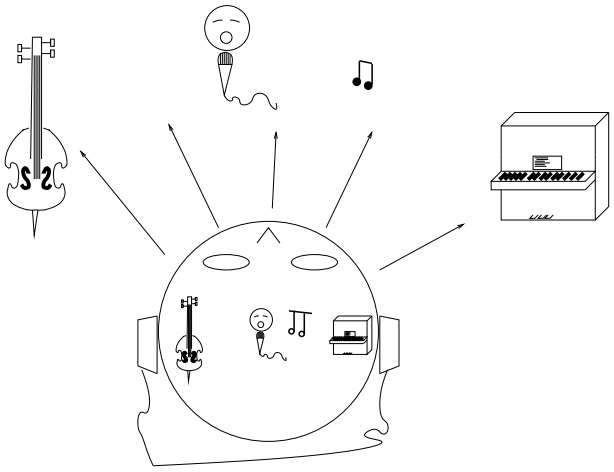


图17：耳机/扬声器外化声源与常规颅内声效对比

需注意耳机/扬声器技术有别于常见的空间处理效果（如回声、混响或低音增强）。 该技术通过精确模拟人类听觉模型及真实听音环境声学特性实现声场外化。因此不会引入人工特效改变音乐的音色特质。

耳机/扬声器处理支持多种参数化模式，每种模式模拟略有差异的声学环境，适配不同个人偏好与录音类型。具体模式激活方式参见章节9.6.1。

- 关闭：适用于扬声器播放或已含双声道预处理的音频内容
- 微调：适配耳机聆听常规乐谱，效果极为精妙

- *normal*: Suited for listening to normal musical scores with headphones, moves sound source further away than *minimal*.
- *extreme*: Suited for old or 'dry' recordings, or if the audio to be played is artificial, for example generated MIDI.

9.4 Serial Data Interface (SDI)

The serial data interface is meant for transferring compressed data for the different decoders of VS1053b.

If the input of the decoder is invalid or it is not received fast enough, analog outputs are automatically muted.

Also several different tests may be activated through SDI as described in Chapter 10.

9.5 Serial Control Interface (SCI)

The serial control interface is compatible with the SPI bus specification. Data transfers are always 16 bits. VS1053b is controlled by writing and reading the registers of the interface.

The main controls of the serial control interface are:

- control of the operation mode, clock, and builtin effects
- access to status information and header data
- receiving encoded data in recording mode
- uploading and controlling user programs

- 标准模式：适用于通过耳机聆听常规乐谱，将声源定位置于比最小化模式更远的距离。
- 极端模式：适用于老旧或'干声'录音，或当播放内容为人造音频时（例如生成的MIDI文件）。

9.4 串行数据接口 (SDI)

该串行数据接口用于向VS1053b的各类解码器传输压缩数据。

若解码器输入数据无效或接收速率不足，模拟输出将自动静音。

此外，可通过SDI激活多种测试功能，详见第10章说明。

9.5 串行控制接口 (SCI)

该串行控制接口兼容SPI总线规范。数据传输始终以16位为单位进行。通过读写接口寄存器实现对VS1053b的控制。

串行控制接口的主要功能包括：

- 运行模式控制、时钟管理及内置效果器调节
- 访问状态信息和头部数据
- 在录音模式下接收编码数据
- 上传和控制用户程序

9.6 SCI Registers

VS1053b sets DREQ low when it detects an SCI operation (this delay is 16 to 40 CLKI cycles depending on whether an interrupt service routine is active) and restores it when it has processed the operation. The duration depends on the operation. If DREQ is low when an SCI operation is performed, it also stays low after SCI operation processing.

If DREQ is high before a SCI operation, do not start a new SCI/SDI operation before DREQ is high again. If DREQ is low before a SCI operation because the SDI can not accept more data, make certain there is enough time to complete the operation before sending another.

SCI registers, prefix SCI_					
Reg	**Type**	**Reset**	**Time**[1]	**Abbrev[bits]**	**Description**
0x0	rw	0x4000[6]	80 CLKI[4]	MODE	Mode control
0x1	rw	0x000C[3]	80 CLKI	STATUS	Status of VS1053b
0x2	rw	0	80 CLKI	BASS	Built-in bass/treble control
0x3	rw	0	1200 XTALI[5]	CLOCKF	Clock freq + multiplier
0x4	rw	0	100 CLKI	DECODE_TIME	Decode time in seconds
0x5	rw	0	450 CLKI[2]	AUDATA	Misc. audio data
0x6	rw	0	100 CLKI	WRAM	RAM write/read
0x7	rw	0	100 CLKI	WRAMADDR	Base address for RAM write/read
0x8	r	0	80 CLKI	HDAT0	Stream header data 0
0x9	r	0	80 CLKI	HDAT1	Stream header data 1
0xA	rw	0	210 CLKI[2]	AIADDR	Start address of application
0xB	rw	0	80 CLKI	VOL	Volume control
0xC	rw	0	80 CLKI[2]	AICTRL0	Application control register 0
0xD	rw	0	80 CLKI[2]	AICTRL1	Application control register 1
0xE	rw	0	80 CLKI[2]	AICTRL2	Application control register 2
0xF	rw	0	80 CLKI[2]	AICTRL3	Application control register 3

[1] This is the worst-case time that DREQ stays low after writing to this register. The user may choose to skip the DREQ check for those register writes that take less than 100 clock cycles to execute and use a fixed delay instead.

[2] In addition, the cycles spent in the user application routine must be counted.

[3] Firmware changes the value of this register immediately to 0x48 (analog enabled), and after a short while to 0x40 (analog drivers enabled).

[4] When mode register write specifies a software reset the worst-case time is 22000 XTALI cycles.

[5] If the clock multiplier is changed, writing to CLOCKF register may force internal clock to run at $1.0 \times$ XTALI for a while. Thus it is not a good idea to send SCI or SDI bits while this register update is in progress.

[6] Firmware changes the value of this register immediately to 0x4800.

Reads from all SCI registers complete in under 100 CLKI cycles, except a read from AIADDR in 200 cycles. In addition the cycles spent in the user application routine must be counted to the read time of AIADDR, AUDATA, and AICTRL0..3.

9.6 SCI寄存器

当检测到SCI操作时，VS1053b会将DREQ置低（此延迟为16至40 CLKI时钟周期，具体取决于中断服务程序是否激活），并在处理完操作后恢复DREQ电平。持续时间取决于具体操作。若执行SCI操作时DREQ为低电平，则在SCI操作处理后仍保持低电平。

若SCI操作前DREQ为高电平，则在DREQ重新变高之前切勿启动新的SCI/SDI操作。若因SDI无法接收更多数据导致SCI操作前DREQ为低电平，请确保发送新操作前留有足够时间完成当前操作。

SCI寄存器，前缀SCI_					
寄存器	类型	复位值	时间[1]	缩写[位数]	描述
0x0	读写	0x4000[6]	80 CLKI[4]	模式寄存器	模式控制
0x1	读写	0x000C[3]	80 CLKI	状态寄存器	VS1053b状态
0x2	读写	0	80 CLKI	低音增强	内置低音/高音控制
0x3	读写	0	1200 XTALI[5]	时钟控制	时钟频率及倍频器
0x4	读写	0	100 CLKI	DECODE_TIME	解码时间（秒）
0x5	读写	0	450 CLKI[2]	AUDATA	其他音频数据
0x6	读写	0	100 CLKI	WRAM	RAM写入/读取
0x7	读写	0	100 CLKI	WRAMADDR	RAM读写基地址
0x8	r	0	80 CLKI	HDAT0	流标头数据0
0x9	r	0	80 CLKI	HDAT1	流标头数据1
0xA	读写	0	210 CLKI[2]	特定术语：AIADDR	应用程序起始地址
0xB	读写	0	80 CLKI	VOL	音量控制
0xC	读写	0	80 CLKI[2]	AICTRL0	应用控制寄存器0
0xD	读写	0	80 CLKI[2]	AICTRL1	应用控制寄存器1
0xE	读写	0	80 CLKI[2]	AICTRL2	应用控制寄存器2
0xF	读写	0	80 CLKI[2]	AICTRL3	应用控制寄存器3

[1] 这是向该寄存器写入数据后，DREQ信号保持低电平的最坏情况持续时间。用户可选择跳过执行时间少于100个时钟周期的寄存器写入操作的DREQ检查，改用固定延迟替代。

[2] 此外，必须计入用户应用程序例程消耗的时钟周期。

[3] 固件会立即将此寄存器值更改为0x48（模拟部分启用），随后在短时间内更改为0x40（模拟驱动器启用）。

[4] 当模式寄存器写入操作指定软件复位时，最坏情况耗时为22000个XTALI时钟周期。

[5] 若更改时钟倍频器，向CLOCKF寄存器写入数据可能导致内部时钟暂时运行于 $1.0 \times$XTALI频率。因此在该寄存器更新期间发送SCI或SDI数据位并非明智之举。

[6] 固件会立即将此寄存器值更改为0x4800。

读取所有SCI寄存器可在100个CLKI时钟周期内完成，但读取AIADDR需要200个周期。此外，用户应用程序例程消耗的周期时间必须计入AIADDR、AUDATA及AICTRL0..3寄存器的读取时间。

9.6.1 SCI_MODE (RW)

SCI_MODE is used to control the operation of VS1053b and defaults to 0x4800 (SM_SDINEW set).

Bit	Name	Function	Value	Description
0	SM_DIFF	Differential	0 1	normal in-phase audio left channel inverted
1	SM_LAYER12	Allow MPEG layers I & II	0 1	no yes
2	SM_RESET	Soft reset	0 1	no reset reset
3	SM_CANCEL	Cancel decoding current file	0 1	no yes
4	SM_EARSPEAKER_LO	EarSpeaker low setting	0 1	off active
5	SM_TESTS	Allow SDI tests	0 1	not allowed allowed
6	SM_STREAM	Stream mode	0 1	no yes
7	SM_EARSPEAKER_HI	EarSpeaker high setting	0 1	off active
8	SM_DACT	DCLK active edge	0 1	rising falling
9	SM_SDIORD	SDI bit order	0 1	MSb first MSb last
10	SM_SDISHARE	Share SPI chip select	0 1	no yes
11	SM_SDINEW	VS1002 native SPI modes	0 1	no yes
12	SM_ADPCM	PCM/ADPCM recording active	0 1	no yes
13	-	-	0 1	right wrong
14	SM_LINE1	MIC / LINE1 selector	0 1	MICP LINE1
15	SM_CLK_RANGE	Input clock range	0 1	12..13 MHz 24..26 MHz

When SM_DIFF is set, the player inverts the left channel output. For a stereo input this creates virtual surround, and for a mono input this creates a differential left/right signal.

SM_LAYER12 enables MPEG 1.0 and 2.0 layer I and II decoding in addition to layer III. **If you enable Layer I and Layer II decoding, you are liable for any patent issues that may arise.** Joint licensing of MPEG 1.0 / 2.0 Layer III does not cover all patents pertaining to layers I and II.

Software reset is initiated by setting SM_RESET to 1. This bit is cleared automatically.

If you want to stop decoding a in the middle, set SM_CANCEL, and continue sending data honouring DREQ. When SM_CANCEL is detected by a codec, it will stop decoding and return to the main loop. The stream buffer content is discarded and the SM_CANCEL bit cleared. SCI_HDAT1 will also be cleared. See Chapter 10.5.2 for details.

9.6.1 SCI_MODE (读写寄存器)

SCI_MODE寄存器用于控制VS1053b的操作，默认值为0x4800（SM_SDINEW置位）。

位域	名称	功能	取值	描述
0	SM_DIFF	差分模式	0 1	常规同相音频 左声道反相
1	SM_LAYER12	允许MPEG I/II层解码	0 1	禁用 启用
2	SM_RESET	软复位	0 1	不复位 复位
3	SM_CANCEL	取消当前文件解码	0 1	禁用 启用
4	SM_EARSPEAKER_LO	耳机/扬声器低电平设置	0 1	关闭 激活
5	SM_TESTS	允许SDI测试	0 1	不允许 允许
6	SM_STREAM	流模式	0 1	禁用 启用
7	SM_EARSPEAKER_HI	耳机/扬声器高电平设置	0 1	关闭 激活
8	SM_DACT	DCLK有效边沿	0 1	上升沿 下降沿
9	SM_SDIORD	SDI位顺序	0 1	最高有效位优先 最低有效位优先
10	SM_SDISHARE	共享SPI片选	0 1	禁用 启用
11	SM_SDINEW	VS1002原生SPI模式	0 1	禁用 启用
12	SM_ADPCM	PCM/ADPCM录音激活	0 1	禁用 启用
13	-	-	0 1	正确 错误
14	SM_LINE1	麦克风或LINE1选择器	0 1	MICP LINE1
15	SM_CLK_RANGE	输入时钟范围	0 1	12..13 兆赫兹 24..26 兆赫兹

当设置SM_DIFF时，播放器会反转左声道输出。对于立体声输入，这将创建虚拟环绕声效果；对于单声道输入，则会生成差分左右声道信号。

SM_LAYER12启用MPEG 1.0和2.0的I层及II层解码（除III层外）。**若启用I层和II层解码，您须自行承担可能产生的专利风险。**
MPEG 1.0/2.0 Layer III的联合许可不涵盖涉及Layer I和II的所有专利。

通过将SM_RESET置1可启动软件复位。该比特位会自动清除。

若需中途停止解码，请设置SM_CANCEL并继续遵守DREQ信号发送数据。当编解码器检测到SM_CANCEL时，将停止解码并返回主循环。流缓冲区内容将被丢弃，且SM_CANCEL比特位自动清除。
SCI_HDAT1寄存器也将被清零。详情参见第10.5.2章。

Bits SM_EARSPEAKER_LO and SM_EARSPEAKER_HI control the EarSpeaker spatial processing. If both are 0, the processing is not active. Other combinations activate the processing and select 3 different effect levels: LO = 1, HI = 0 selects *minimal*, LO = 0, HI = 1 selects *normal*, and LO = 1, HI = 1 selects *extreme*. EarSpeaker takes approximately 12 MIPS at 44.1 kHz samplerate.

If SM_TESTS is set, SDI tests are allowed. For more details on SDI tests, look at Chapter 10.12.

SM_STREAM activates VS1053b's stream mode. In this mode, data should be sent with as even intervals as possible and preferable in blocks of less than 512 bytes, and VS1053b makes every attempt to keep its input buffer half full by changing its playback speed upto 5%. For best quality sound, the average speed error should be within 0.5%, the bitrate should not exceed 160 kbit/s and VBR should not be used. For details, see Application Notes for VS10XX. This mode only works with MP3 and WAV files.

SM_DACT defines the active edge of data clock for SDI. When '0', data is read at the rising edge, when '1', data is read at the falling edge.

When SM_SDIORD is clear, bytes on SDI are sent MSb first. By setting SM_SDIORD, the user may reverse the bit order for SDI, i.e. bit 0 is received first and bit 7 last. Bytes are, however, still sent in the default order. This register bit has no effect on the SCI bus.

Setting SM_SDISHARE makes SCI and SDI share the same chip select, as explained in Chapter 7.1, if also SM_SDINEW is set.

Setting SM_SDINEW will activate VS1002 native serial modes as described in Chapters 7.1.1 and 7.3.1. Note, that this bit is set as a default when VS1053b is started up.

By activating SM_ADPCM and SM_RESET at the same time, the user will activate IMA ADPCM recording mode (see section 10.8).

SM_LINE_IN is used to select the left-channel input for ADPCM recording. If '0', differential microphone input pins MICP and MICN are used; if '1', line-level MICP/LINEIN1 pin is used.

SM_CLK_RANGE activates a clock divider in the XTAL input. When SM_CLK_RANGE is set, the clock is divided by 2 at the input. From the chip's point of view e.g. 24 MHz becomes 12 MHz. SM_CLK_RANGE should be set as soon as possible after a chip reset.

寄存器位 SM_EARSPEAKER_LO 和 SM_EARSPEAKER_HI 控制 EarSpeaker 空间处理功能。若两者均为0，则处理功能不启用。其他组合将激活处理功能并提供3种效果级别选择：LO=1, HI=0 选择最小化效果；LO=0, HI=1 选择常规效果；LO=1, HI=1 选择极致效果。EarSpeaker 功能在 44.1 kHz 采样率下约占用 12 MIPS 运算资源。

若设置 SM_TESTS 标志，则允许执行 SDI 测试。SDI 测试的详细说明请参阅第 10.12 章节。

SM_STREAM 标志激活 VS1053b 的流传输模式。此模式下，数据应尽可能以均匀间隔传输，建议采用小于 512 字节的数据块；VS1053b 将通过最高 ±5% 的播放速度调节，竭力维持其输入缓冲区处于半满状态。为获得最佳音质，平均速度误差应控制在0.5%以内，比特率不得超过160 kbit/s，且不应使用VBR（可变比特率）。 详情请参阅《VS10XX 系列应用说明》。此模式仅适用于MP3和WAV音频文件。

SM_DACT寄存器定义SDI数据时钟的有效边沿：设为'0'时在上升沿读取数据，设为'1'时在下降沿读取数据。

当SM_SDIORD清零时，SDI总线上的字节按最高有效位（MSb）优先顺序发送。通过设置SM_SDIORD，用户可反转SDI的位传输顺序：即先传输bit 0，最后传输bit 7。 但字节仍按默认顺序发送。该寄存器位对SCI总线无影响。

设置SM_SDISHARE可使SCI与SDI共享同一片选信号（需同时设置SM_SDINEW），具体说明详见第7.1章。

设置SM_SDINEW将激活VS1002原生串行模式，具体说明见第7.1.1章与第7.3.1章。
需注意，VS1053b启动时默认设置该位。

同时激活SM_ADPCM和SM_RESET将启用IMA ADPCM录音模式（参见第10.8节）。

SM_LINE_IN用于选择ADPCM录音的左声道输入源。若为'0'，则使用差分麦克风输入引脚MICP和MICN； 若为'1'，则使用线路电平MICP/LINEIN1引脚。

SM_CLK_RANGE激活XTAL输入端的时钟分频器。设置SM_CLK_RANGE后，输入时钟将进行二分频。从芯片视角看，例如24MHz将变为12MHz。芯片复位后应尽快设置SM_CLK_RANGE。

9.6.2 SCI_STATUS (RW)

SCI_STATUS contains information on the current status of VS1053b. It also controls some low-level things that the user does not usually have to care about.

Name	Bits	Description
SS_DO_NOT_JUMP	15	Header in decode, do not fast forward/rewind
SS_SWING	14:12	Set swing to +0 dB, +0.5 dB, .., or +3.5 dB
SS_VCM_OVERLOAD	11	GBUF overload indicator '1' = overload
SS_VCM_DISABLE	10	GBUF overload detection '1' = disable
	9:8	reserved
SS_VER	7:4	Version
SS_APDOWN2	3	Analog driver powerdown
SS_APDOWN1	2	Analog internal powerdown
SS_AD_CLOCK	1	AD clock select, '0' = 6 MHz, '1' = 3 MHz
SS_REFERENCE_SEL	0	Reference voltage selection, '0' = 1.23 V, '1' = 1.65 V

SS_DO_NOT_JUMP is set when a WAV, Ogg Vorbis, WMA, MP4, or AAC-ADIF header is being decoded and jumping to another location in the file is not allowed. If you use soft reset or cancel, clear this bit yourself or it can be accidentally left set.

If AVDD is at least 3.3 V, SS_REFERENCE_SEL can be set to select 1.65 V reference voltage to increase the analog output swing.

SS_AD_CLOCK can be set to divide the AD modulator frequency by 2 if XTALI/2 is too much.

SS_VER is 0 for VS1001, 1 for VS1011, 2 for VS1002, 3 for VS1003, 4 for VS1053 and VS8053, 5 for VS1033, 7 for VS1103, and 6 for VS1063.

SS_APDOWN2 controls analog driver powerdown. SS_APDOWN1 controls internal analog powerdown. These bit are meant to be used by the system firmware only.

If the user wants to powerdown VS1053b with a minimum power-off transient, set SCI_VOL to 0xffff, then wait for at least a few milliseconds before activating reset.

VS1053b contains GBUF protection circuit which disconnects the GBUF driver when too much current is drawn, indicating a short-circuit to ground. SS_VCM_OVERLOAD is high while the overload is detected. SS_VCM_DISABLE can be set to disable the protection feature.

SS_SWING allows you to go above the 0 dB volume setting. Value 0 is normal mode, 1 gives +0.5 dB, and 2 gives +1.0 dB. Settings from 3 to 7 cause the DAC modulator to be overdriven and should not be used. You can use SS_SWING with I2S to control the amount of headroom.

Note: Due to a firmware bug in the VS1053b volume calculation routine clears SS_AD_CLOCK and SS_REFERENCE_SEL bits. Write to SCI_STATUS or SCI_VOLUME, and sample rate change (if bass enhancer or treble control are active) causes the volume calculation routine to be called. See the *VS1053b Patches w/ FLAC Decoder* plugin for a workaround:
http://www.vlsi.fi/en/support/software/vs10xxplugins.html

9.6.2 SCI_STATUS寄存器（可读写）

SCI_STATUS寄存器包含VS1053b当前状态信息，同时控制用户通常无需关注的底层功能。

名称	位	描述
SS_DO_NOT_JUMP	15	解码头文件时禁止快进/快退
SS_SWING	14:12	将摆幅设置为+0 dB、+0.5 dB、...或+3.5 dB
SS_VCM_OVERLOAD	11	GBUF过载指示符：'1'=过载
SS_VCM_DISABLE	10	GBUF过载检测：'1'=禁用
	9:8	保留
SS_VER	7:4	版本
SS_APDOWN2	3	模拟驱动器断电
SS_APDOWN1	2	模拟内部电路断电
SS_AD_CLOCK	1	AD时钟选择：'0'=6兆赫，'1'=3兆赫
SS_REFERENCE_SEL	0	参考电压选择：'0'=1.23伏，'1'=1.65伏

当解码WAV、Ogg Vorbis、WMA、MP4或AAC-ADIF头文件时，SS_DO_NOT_JUMP标志位会被置位，此时禁止跳转至文件其他位置。若执行软复位或取消操作，需手动清除该位，否则可能意外保持置位状态。

若AVDD电压不低于3.3V，可设置SS_REFERENCE_SEL选择1.65V参考电压以增大模拟输出摆幅。

当XTALI/2频率过高时，可设置SS_AD_CLOCK将AD调制器频率二分频。

SS_VER值定义：0对应VS1001，1对应VS1011，2对应VS1002，3对应VS1003，4对应VS1053和VS8053，5对应VS1033，7对应VS1103，6对应VS1063。

SS_APDOWN2寄存器控制模拟驱动器掉电功能。SS_APDOWN1寄存器控制内部模拟电路掉电功能。这些比特位仅供系统固件使用。

若需以最小断电瞬态关闭VS1053b，应先将SCI_VOL设为0xffff，等待至少数毫秒后再激活复位信号。

VS1053b内置GBUF保护电路，当电流过大时（表明对地短路）将断开GBUF驱动器。检测到过载时，SS_VCM_OVERLOAD信号将保持高电平。可通过设置SS_VCM_DISABLE禁用该保护功能。

SS_SWING功能允许突破0 dB音量设置上限。值0为正常模式，1提升+0.5 dB，2提升+1.0 dB。设置值3至7将导致DAC调制器过载，不应使用。可通过SS_SWING配合I2S接口控制动态余量幅度。

注意：因VS1053b音量计算固件存在缺陷，写入操作会清空SS_AD_CLOCK和SS_REFERENCE_SEL寄存器位。写入SCI_STATUS或SCI_VOLUME寄存器时，若低音增强器或高音控制功能处于激活状态，采样率变化将触发音量计算程序。请参阅*VS1053b*补丁与*FLAC*解码器插件获取解决方案：
http://www.vlsi.fi/en/support/software/vs10xxplugins.html

9.6.3 SCI_BASS (RW)

Name	Bits	Description
ST_AMPLITUDE	15:12	Treble Control in 1.5 dB steps (-8..7, 0 = off)
ST_FREQLIMIT	11:8	Lower limit frequency in 1000 Hz steps (1..15)
SB_AMPLITUDE	7:4	Bass Enhancement in 1 dB steps (0..15, 0 = off)
SB_FREQLIMIT	3:0	Lower limit frequency in 10 Hz steps (2..15)

The Bass Enhancer VSBE is a powerful bass boosting DSP algorithm, which tries to take the most out of the users earphones without causing clipping.

VSBE is activated when SB_AMPLITUDE is non-zero. SB_AMPLITUDE should be set to the user's preferences, and SB_FREQLIMIT to roughly 1.5 times the lowest frequency the user's audio system can reproduce. For example setting SCI_BASS to 0x00f6 will have 15 dB enhancement below 60 Hz.

Note: Because VSBE tries to avoid clipping, it gives the best bass boost with dynamical music material, or when the playback volume is not set to maximum. It also does not create bass: the source material must have some bass to begin with.

Treble Control VSTC is activated when ST_AMPLITUDE is non-zero. For example setting SCI_BASS to 0x7a00 will have 10.5 dB treble enhancement at and above 10 kHz.

Bass Enhancer uses about 2.1 MIPS and Treble Control 1.2 MIPS at 44100 Hz samplerate. Both can be on simultaneously.

In VS1053b bass and treble initialization and volume change is delayed until the next batch of samples are sent to the audio FIFO. Thus, unlike with earlier VS10XX chips, audio interrupts can no longer be missed when SCI_BASS or SCI_VOL is written to.

9.6.3 SCI_BASS寄存器（可读写）

名称	位	描述
ST_AMPLITUDE	15:12	高音控制：1.5 dB步进调节（-8..7，0=关闭）
ST_FREQLIMIT	11:8	下限频率：1000赫兹步进设置（1..15）
SB_AMPLITUDE	7:4	低音增强：1 dB步进调节（0..15，0=关闭）
SB_FREQLIMIT	3:0	下限频率：10赫兹步进设置（2..15）

低音增强器VSBE是一种强大的低音提升DSP算法，旨在最大限度发挥用户耳机的低频表现而不引发削波失真。

当SB_AMPLITUDE寄存器值非零时，VSBE功能被激活。SB_AMPLITUDE应依据用户偏好设置，SB_FREQLIMIT需设置为用户音频系统可重现最低频率的约1.5倍。例如：将SCI_BASS设为0x00f6时，将在60赫兹以下频段实现15 dB增强效果。

注意：由于低音增强器(VSBE)会避免削波失真，因此它在播放动态音乐素材或未将音量调至最大时能提供最佳的低音增强效果。它本身不会产生低音：音源素材必须本身包含低音成分才能发挥作用。

当ST_AMPLITUDE值为非零时，高音控制(VSTC)功能将被激活。例如：将SCI_BASS寄存器设为0x7a00时，在10千赫兹及以上频段可获得10.5分贝的高音增强效果。

在44100赫兹采样率下，低音增强器约消耗2.1 MIPS运算资源，高音控制功能消耗1.2 MIPS。两项功能可同时启用。

在VS1053b中，低音/高音初始化及音量变更操作将延迟至下一批样本送入音频FIFO后才生效。因此与早期VS10XX芯片不同，写入SCI_BASS或SCI_VOL寄存器时不会再导致音频中断丢失。

9.6.4 SCI_CLOCKF (RW)

The external clock multiplier SCI register SCI_CLOCKF, which has changed slightly since VS1003 and VS1033, is presented in the table below.

SCI_CLOCKF bits		
Name	**Bits**	**Description**
SC_MULT	15:13	Clock multiplier
SC_ADD	12:11	Allowed multiplier addition
SC_FREQ	10: 0	Clock frequency

SC_MULT activates the built-in clock multiplier. This will multiply XTALI to create a higher CLKI. When the multiplier is changed by more than $0.5\times$, the chip runs at $1.0\times$ clock for a few hundres clock cycles. The values are as follows:

SC_MULT	MASK	CLKI
0	0x0000	XTALI
1	0x2000	XTALI$\times 2.0$
2	0x4000	XTALI$\times 2.5$
3	0x6000	XTALI$\times 3.0$
4	0x8000	XTALI$\times 3.5$
5	0xa000	XTALI$\times 4.0$
6	0xc000	XTALI$\times 4.5$
7	0xe000	XTALI$\times 5.0$

SC_ADD tells how much the decoder firmware is allowed to add to the multiplier specified by SC_MULT if more cycles are temporarily needed to decode a WMA or AAC stream. The values are:

SC_ADD	MASK	Multiplier addition
0	0x0000	No modification is allowed
1	0x0800	XTALI$\times 1.0$
2	0x1000	XTALI$\times 1.5$
3	0x1800	XTALI$\times 2.0$

If SC_FREQ is non-zero, it tells that the input clock XTALI is running at something else than 12.288 MHz. XTALI is set in 4 kHz steps. The formula for calculating the correct value for this register is $\frac{XTALI-8000000}{4000}$ (XTALI is in Hz).

Note: because maximum samplerate is $\frac{XTALI}{256}$, all samplerates are not available if XTALI $<$ 12.288 MHz.

Note: Automatic clock change can only happen when decoding WMA and AAC files. Automatic clock change is done one $0.5\times$ at a time. This does not cause a drop to $1.0\times$ clock and you can use the same SCI and SDI clock throughout the file.

Example: If SCI_CLOCKF is 0x8BE8, SC_MULT = 4, SC_ADD = 1 and SC_FREQ = 0x3E8 = 1000. This means that XTALI = $1000 \times 4000 + 8000000$ = 12 MHz. The clock multiplier is set to $3.5\times$XTALI $= 42$ MHz, and the maximum allowed multiplier that the firmware may automatically choose to use is $(3.5 + 1.0)\times$XTALI $= 54$ MHz.

9.6.4 SCI_CLOCKF寄存器(读写)

外部时钟倍频器SCI寄存器SCI_CLOCKF（自VS1003和VS1033以来略有变化）如下表所示。

SCI_CLOCKF寄存器位域		
名称	**位**	**描述**
SC_MULT	15:13	时钟倍频系数
SC_ADD	12:11	允许的倍频增量
SC_FREQ	10: 0	时钟频率

SC_MULT寄存器用于激活内置时钟倍频器。该功能将倍增XTALI输入以产生更高频率的CLKI时钟。当倍频系数变化超过 0.5 ×时，芯片将暂时以 $1.0\times$ 倍时钟频率运行数百个时钟周期。具体取值如下：

SC_MULT	掩码值	特定术语：CLKI
0	0x0000	XTALI
1	0x2000	XTALI$\times 2.0$
2	0x4000	XTALI$\times 2.5$
3	0x6000	XTALI$\times 3.0$
4	0x8000	XTALI$\times 3.5$
5	0xa000	XTALI$\times 4.0$
6	0xc000	XTALI$\times 4.5$
7	0xe000	XTALI$\times 5.0$

SC_ADD寄存器定义了当解码WMA或AAC流临时需要更多时钟周期时，解码器固件允许在SC_MULT寄存器指定的倍频器基础上增加的增量。取值如下：

SC_ADD	掩码值	倍频器增量
0	0x0000	不允许修改
1	0x0800	XTALI$\times 1.0$
2	0x1000	XTALI$\times 1.5$
3	0x1800	XTALI$\times 2.0$

若SC_FREQ非零，则表明输入时钟XTALI未运行于12.288 MHz。XTALI以4 kHz为步进单位设置。该寄存器的正确值计算公式为$\frac{XTALI-8000000}{4000}$ (XTALI单位为赫兹)。

注意：因最高采样率为$\frac{XTALI}{256}$，故当XTALI $<$ 12.288 兆赫兹时，并非所有采样率都可用。

注意：仅当解码WMA和AAC文件时才会触发自动时钟切换。自动时钟切换每次以 0.5 × 为步进单位完成。该操作不会导致时钟降至 $1.0\times$ 且全程可使用相同的SCI与SDI时钟处理整个文件。

示例：若SCI_CLOCKF寄存器值为0x8BE8，则SC_MULT = 4、SC_ADD = 1、SC_FREQ = 0x3E8 = 1000。这意味着XTALI晶振频率 = $1000\times 4000+8000000$= 12兆赫。时钟倍频器设置为$3.5\times$XTALI$=42$兆赫，固件可自动选择的最大允许倍频值为 $(3.5+1.0)\times$XTALI$=54$兆赫。

9.6.5 SCI_DECODE_TIME (RW)

When decoding correct data, current decoded time is shown in this register in full seconds.

The user may change the value of this register. In that case the new value should be written twice to make absolutely certain that the change is not overwritten by the firmware. A write to SCI_DECODE_TIME also resets the `byteRate` calculation.

SCI_DECODE_TIME is reset at every hardware and software reset. It is no longer cleared when decoding of a file ends to allow the decode time to proceed automatically with looped files and with seamless playback of multiple files.

With fast playback (see the `playSpeed` extra parameter) the decode time also counts faster.

Some codecs (WMA and Ogg Vorbis) can also indicate the absolute play position, see the `positionMsec` extra parameter in section 10.11.

9.6.6 SCI_AUDATA (RW)

When decoding correct data, the current samplerate and number of channels can be found in bits 15:1 and 0 of SCI_AUDATA, respectively. Bits 15:1 contain the samplerate divided by two, and bit 0 is 0 for mono data and 1 for stereo. Writing to SCI_AUDATA will change the samplerate directly.

Example: 44100 Hz stereo data reads as 0xAC45 (44101).
Example: 11025 Hz mono data reads as 0x2B10 (11024).
Example: Writing 0xAC80 sets samplerate to 44160 Hz, stereo mode does not change.

To reduce digital power consumption when idle, you can write a low samplerate to SCI_AUDATA.

Note: Ogg Vorbis decoding overrides AUDATA change. If you want to fine-tune samplerate in streaming applications with Ogg Vorbis, use SCI_CLOCKF to control the playback rate instead of AUDATA.

9.6.7 SCI_WRAM (RW)

SCI_WRAM is used to upload application programs and data to instruction and data RAMs. The start address must be initialized by writing to SCI_WRAMADDR prior to the first write/read of SCI_WRAM. As 16 bits of data can be transferred with one SCI_WRAM write/read, and the instruction word is 32 bits long, two consecutive writes/reads are needed for each instruction word. The byte order is big-endian (i.e. most significant words first). After each full-word write/read, the internal pointer is autoincremented.

9.6.8 SCI_WRAMADDR (W)

SCI_WRAMADDR is used to set the program address for following SCI_WRAM writes/reads. Use an address offset from the following table to access X, Y, I or peripheral memory.

9.6.5 SCI_DECODE_TIME (读写)

解码正确数据时，此寄存器以整秒形式显示当前解码时间。

用户可更改此寄存器的数值。此时新值需写入两次，以确保该变更不会被固件覆盖。写入 SCI_DECODE_TIME 寄存器也会重置字节率(byteRate)计算。

SCI_DECODE_TIME 会在每次硬件复位和软件复位时被重置。文件解码结束时不再清除该值，使得循环文件及多文件无缝播放时解码时间可自动累计。

快放模式下（参见播放速度(playSpeed)额外参数），解码时间计数也会相应加快。

部分编解码器（WMA 和 Ogg Vorbis 格式）可反馈绝对播放位置，详见 10.11 节的 positionMsec 额外参数。

9.6.6 SCI_AUDATA (读写寄存器)

解码正确数据时，当前采样率(samplerate)和声道数分别位于 SCI_AUDATA 寄存器的位[15:1]和位[0]。 位15至1包含采样率除以二的值，位0为0表示单声道数据，为1表示立体声数据。向SCI_AUDATA寄存器写入数据将直接改变采样率。

示例：44100赫兹立体声数据读取值为0xAC45（对应十进制44101）。
示例：11025赫兹单声道数据读取值为0x2B10（对应十进制11024）。
示例：写入0xAC80将采样率设置为44160赫兹，立体声模式保持不变。

为降低空闲时的数字功耗，可向SCI_AUDATA写入低采样率值。

注意：Ogg Vorbis格式解码会覆盖AUDATA的修改。若需在Ogg Vorbis流媒体应用中微调采样率，应使用SCI_CLOCKF寄存器控制播放速率而非AUDATA。

9.6.7 SCI_WRAM（读写）

SCI_WRAM用于向指令RAM和数据RAM上传应用程序及数据。
起始地址必须通过向SCI_WRAMADDR写入进行初始化，然后才能首次读写SCI_WRAM。由于每次SCI_WRAM写入/读取可传输16位数据，而指令字长为32位，因此每个指令字需要执行两次连续的写入/读取操作。字节序采用大端模式（即最高有效字在前）。 每次完成全字写入/读取后，内部指针会自动递增。

9.6.8 SCI_WRAMADDR (W)

SCI_WRAMADDR用于设置后续SCI_WRAM写入/读取操作的程序地址。
使用下表中的地址偏移量访问X、Y、I或外设存储器。

WRAMADDR Start...End	Dest. addr. Start...End	Bits/ Word	Description
0x1800...0x18XX	0x1800...0x18XX	16	X data RAM
0x5800...0x58XX	0x1800...0x18XX	16	Y data RAM
0x8040...0x84FF	0x0040...0x04FF	32	Instruction RAM
0xC000...0xFFFF	0xC000...0xFFFF	16	I/O

Only user areas in X, Y, and instruction memory are listed above. Other areas can be accessed, but should not be written to unless otherwise specified.

9.6.9 SCI_HDAT0 and SCI_HDAT1 (R)

For WAV files, SCI_HDAT1 contains 0x7665 (“ve”). SCI_HDAT0 contains the data rate measured in bytes per second for all supported RIFF WAVE formats: mono and stereo 8-bit or 16-bit PCM, mono and stereo IMA ADPCM. To get the bitrate of the file, multiply the value by 8.

For AAC ADTS streams, SCI_HDAT1 contains 0x4154 (“AT”). For AAC ADIF files, SCI_HDAT1 contains 0x4144 (“AD”). For AAC .mp4 / .m4a files, SCI_HDAT1 contains 0x4D34 (“M4”). SCI_HDAT0 contains the average data rate in bytes per second. To get the bitrate of the file, multiply the value by 8.

For WMA files, SCI_HDAT1 contains 0x574D (“WM”) and SCI_HDAT0 contains the data rate measured in bytes per second. To get the bitrate of the file, multiply the value by 8.

For MIDI files, SCI_HDAT1 contains 0x4D54 (“MT”) and SCI_HDAT0 contains the average data rate in bytes per second. To get the bitrate of the file, multiply the value by 8.

For Ogg Vorbis files, SCI_HDAT1 contains 0x4F67 “Og”. SCI_HDAT0 contains the average data rate in bytes per second. To get the bitrate of the file, multiply the value by 8.

For MP3 files, SCI_HDAT1 is between 0xFFE0 and 0xFFFF. SCI_HDAT1 / 0 contain the following:

WRAMADDR 起始...结束	目标地址 起始...结束	位/ 字长	描述
0x1800. . . 0x18XX	0x1800. . . 0x18XX	16	X数据RAM
0x5800. . . 0x58XX	0x1800. . . 0x18XX	16	Y数据RAM
0x8040. . . 0x84FF	0x0040. . . 0x04FF	32	指令RAM
0xC000. . . 0xFFFF	0xC000. . . 0xFFFF	16	I/O

上方仅列出X、Y及指令存储器中的用户可访问区域。其他区域可被访问，但除非另有说明，否则不应进行写入操作。

9.6.9 SCI_HDAT0与SCI_HDAT1寄存器（只读）

对于WAV文件，SCI_HDAT1寄存器包含0x7665（“ve”）。 对于所有支持的RIFF WAV格式（单声道/立体声8位或16位PCM、单声道/立体声IMA ADPCM），SCI_HDAT0寄存器包含以字节/秒为单位的数据速率。获取文件比特率需将该值乘以8。

对于AAC ADTS流，SCI_HDAT1寄存器包含0x4154（“AT”）。 对于AAC ADIF文件，SCI_HDAT1寄存器包含0x4144（“AD”）。 对于AAC .mp4/.m4a文件，SCI_HDAT1寄存器包含0x4D34（“M4”）。SCI_HDAT0寄存器包含以字节/秒为单位的平均数据速率。要获取文件比特率，需将该值乘以8。

对于WMA文件，SCI_HDAT1寄存器包含0x574D（“WM”），SCI_HDAT0寄存器包含以字节/秒为单位测量的数据速率。要获取文件比特率，需将该值乘以8。

对于MIDI文件，SCI_HDAT1寄存器包含0x4D54（“MT”），SCI_HDAT0寄存器包含以字节/秒为单位的平均数据速率。要获取文件比特率，需将该值乘以8。

对于Ogg Vorbis文件，SCI_HDAT1寄存器包含0x4F67（“Og”）。 SCI_HDAT0寄存器包含以字节/秒为单位的平均数据速率。要获取文件比特率，需将该值乘以8。

对于MP3文件，SCI_HDAT1寄存器的值介于0xFFE0至0xFFFF之间。SCI_HDAT1/0寄存器包含以下内容：

Bit	Function	Value	Explanation
HDAT1[15:5]	syncword	2047	stream valid
HDAT1[4:3]	ID	3	ISO 11172-3 MPG 1.0
		2	ISO 13818-3 MPG 2.0 (1/2-rate)
		1	MPG 2.5 (1/4-rate)
		0	MPG 2.5 (1/4-rate)
HDAT1[2:1]	layer	3	I
		2	II
		1	III
		0	reserved
HDAT1[0]	protect bit	1	No CRC
		0	CRC protected
HDAT0[15:12]	bitrate		see bitrate table
HDAT0[11:10]	samplerate	3	reserved
		2	32/16/ 8 kHz
		1	48/24/12 kHz
		0	44/22/11 kHz
HDAT0[9]	pad bit	1	additional slot
		0	normal frame
HDAT0[8]	private bit		not defined
HDAT0[7:6]	mode	3	mono
		2	dual channel
		1	joint stereo
		0	stereo
HDAT0[5:4]	extension		see ISO 11172-3
HDAT0[3]	copyright	1	copyrighted
		0	free
HDAT0[2]	original	1	original
		0	copy
HDAT0[1:0]	emphasis	3	CCITT J.17
		2	reserved
		1	50/15 microsec
		0	none

When read, SCI_HDAT0 and SCI_HDAT1 contain header information that is extracted from MP3 stream currently being decoded. After reset both registers are cleared, indicating no data has been found yet.

The “samplerate” field in SCI_HDAT0 is interpreted according to the following table:

“samplerate”	ID=3	ID=2	ID=0,1
3	-	-	-
2	32000	16000	8000
1	48000	24000	12000
0	44100	22050	11025

The “bitrate” field in HDAT0 is read according to the following table. Notice that for variable bitrate stream the value changes constantly.

位域	功能	取值	说明
HDAT1[15:5]	同步字	2047	流有效
HDAT1[4:3]	ID	3 2 1 0	ISO 11172-3 MPG 1.0 ISO 13818-3 MPG 2.0 (1/2速率) MPG 2.5 (1/4速率) MPG 2.5 (1/4速率)
HDAT1[2:1]	层级	3 2 1 0	I II III 保留
HDAT1[0]	保护位	1 0	无CRC校验 CRC保护
HDAT0[15:12]	特定术语：比特率		参见比特率表
HDAT0[11:10]	采样率	3 2 1 0	保留 32/16/8 千赫兹 48/24/12 千赫兹 44/22/11 千赫兹
HDAT0[8]" (寄存器位保留英文格式)	填充位	1 0	附加时隙 常规帧
HDAT0[8]	私有位		未定义
HDAT0[7:6]	模式	3 2 1 0	单声道 双通道 联合立体声 立体声
HDAT0[5:4]	扩展		参见ISO 11172-3标准
HDAT0[3]	版权标识	1 0	受版权保护 自由
HDAT0[2]	原始介质	1 0	原始介质 复制
HDAT0[1:0]	加重处理	3 2 1 0	CCITT J.17 保留 50/15微秒 无

读取时，SCI_HDAT0和SCI_HDAT1寄存器包含从当前正在解码的MP3流中提取的帧头信息复位后两个寄存器均清零，表明尚未检测到有效数据

SCI_HDAT0中的「samplerate」字段按以下表格解析：

「samplerate」	ID=3	ID=2	ID=0,1
3	-	-	-
2	32000	16000	8000
1	48000	24000	12000
0	44100	22050	11025

HDAT0中的「bitrate」字段按以下表格读取注意：可变比特率流的该值会不断变化

"bitrate"	Layer I		Layer II		Layer III	
	ID=3	ID=0,1,2	ID=3	ID=0,1,2	ID=3	ID=0,1,2
	kbit/s		kbit/s		kbit/s	
15	forbidden	forbidden	forbidden	forbidden	forbidden	forbidden
14	448	256	384	160	320	160
13	416	224	320	144	256	144
12	384	192	256	128	224	128
11	352	176	224	112	192	112
10	320	160	192	96	160	96
9	288	144	160	80	128	80
8	256	128	128	64	112	64
7	224	112	112	56	96	56
6	192	96	96	48	80	48
5	160	80	80	40	64	40
4	128	64	64	32	56	32
3	96	56	56	24	48	24
2	64	48	48	16	40	16
1	32	32	32	8	32	8
0	-	-	-	-	-	-

The average data rate in bytes per second can be read from memory, see the `byteRate` extra parameter. This variable contains the byte rate for all codecs. To get the bitrate of the file, multiply the value by 8.

The bitrate calculation is not automatically reset between songs, but it can also be reset without a software or hardware reset by writing to SCI_DECODE_TIME.

9.6.10 SCI_AIADDR (RW)

SCI_AIADDR indicates the start address of the application code written earlier with SCI_WRAMADDR and SCI_WRAM registers. If no application code is used, this register should not be initialized, or it should be initialized to zero. For more details, see Application Notes for VS10XX.

Note: Reading AIADDR is not recommended. It can cause samplerate to be set to a very low value.

「bitrate」	Layer I		Layer II		Layer III	
	ID=3	ID=0,1,2	ID=3	ID=0,1,2	ID=3	ID=0,1,2
	kbit/s		kbit/s		kbit/s	
15	禁用	禁用	禁用	禁用	禁用	禁用
14	448	256	384	160	320	160
13	416	224	320	144	256	144
12	384	192	256	128	224	128
11	352	176	224	112	192	112
10	320	160	192	96	160	96
9	288	144	160	80	128	80
8	256	128	128	64	112	64
7	224	112	112	56	96	56
6	192	96	96	48	80	48
5	160	80	80	40	64	40
4	128	64	64	32	56	32
3	96	56	56	24	48	24
2	64	48	48	16	40	16
1	32	32	32	8	32	8
0	-	-	-	-	-	-

每秒平均数据速率（字节/秒）可从内存读取，参见 byteRate额外参数此变量包含所有编解码器的字节速率要获取文件比特率，需将该值乘以8。

比特率计算不会在歌曲切换时自动重置，但可通过写入SCI_DECODE_TIME寄存器实现无需软硬件重置的复位

9.6.10 SCI_AIADDR (RW)

SCI_AIADDR寄存器指示通过SCI_WRAMADDR和SCI_WRAM寄存器之前写入的应用程序代码起始地址。若未使用应用程序代码，则不应初始化该寄存器，或应将其初始化为零。更多细节请参阅《VS10XX系列应用说明》。

注意：不建议读取AIADDR。 这可能导致采样率被设置为极低值。

9.6.11 SCI_VOL (RW)

SCI_VOL is a volume control for the player hardware. The most significant byte of the volume register controls the left channel volume, the low part controls the right channel volume. The channel volume sets the attenuation from the maximum volume level in 0.5 dB steps. Thus, maximum volume is 0x0000 and total silence is 0xFEFE.

Note, that after hardware reset the volume is set to full volume. Resetting the software does not reset the volume setting.

Setting SCI_VOL to 0xFFFF will activate analog powerdown mode.

Example: for a volume of -2.0 dB for the left channel and -3.5 dB for the right channel: (2.0/0.5) = 4, 3.5/0.5 = 7 $\rightarrow$ SCI_VOL = 0x0407.

Example: SCI_VOL = 0x2424 $\rightarrow$ both left and right volumes are 0x24 * -0.5 = -18.0 dB

In VS1053b bass and treble initialization and volume change is delayed until the next batch of samples are sent to the audio FIFO. Thus, audio interrupts can no longer be missed during a write to SCI_BASS or SCI_VOL.

This delays the volume setting slightly, but because the volume control is now done in the DAC hardware instead of performing it to the samples going into the audio FIFO, the overall volume change response is better than before. Also, the actual volume control has zero-cross detection, which almost completely removes all audible noise that occurs when volume is suddenly changed.

9.6.12 SCI_AICTRL[x] (RW)

SCI_AICTRL[x] registers (x=[0 .. 3]) can be used to access the user's application program.

The AICTRL registers are also used with PCM/ADPCM encoding mode.

9.6.11 SCI_VOL (RW)

SCI_VOL是播放器硬件的音量控制寄存器。音量寄存器的高字节控制左声道音量，低字节控制右声道音量。声道音量以0.5dB为步进设置相对于最大音量的衰减量。因此，0x0000表示最大音量，0xFEFE表示完全静音。

注意：硬件复位后音量将设为最大值。软件复位不会重置音量设置。

将SCI_VOL设为0xFFFF将激活模拟掉电模式。

示例：若左声道音量设为-2.0 dB，右声道设为-3.5 dB：(2.0/0.5) = 4，3.5/0.5 = 7 →SCI_VOL = 0x0407。

示例：SCI_VOL = 0x2424 →左右声道音量均为 0x24 * -0.5 = -18.0 分贝

在VS1053b中，低音/高音初始化及音量变更将延迟至下一批样本送入音频FIFO后生效。因此，写入SCI_BASS或SCI_VOL寄存器时不会再错过音频中断。

这会使音量设置产生轻微延迟，但由于音量控制现由DAC硬件执行（而非对输入音频FIFO的样本进行处理），整体音量调节响应速度优于以往。此外，实际音量控制具备过零检测功能，可基本消除音量突变时产生的所有可闻噪声。

9.6.12 SCI_AICTRL[x] (RW)

SCI_AICTRL[x]寄存器（x=[0..3]）可用于访问用户应用程序。

AICTRL寄存器同样适用于PCM/ADPCM编码模式。" // 注：ADPCM按术语表保留

10 Operation

10.1 Clocking

VS1053b operates on a single, nominally 12.288 MHz fundamental frequency master clock. This clock can be generated by external circuitry (connected to pin XTALI) or by the internal clock crystal interface (pins XTALI and XTALO). This clock is used by the analog parts and determines the highest available samplerate. With 12.288 MHz clock all samplerates upto 48000 Hz are available.

VS1053b can also use 24..26 MHz clocks when SM_CLK_RANGE in the SCI_MODE register is set to 1. The system clock is then divided by 2 at the clock input and the chip gets a 12..13 MHz input clock.

10.2 Hardware Reset

When the XRESET -signal is driven low, VS1053b is reset and all the control registers and internal states are set to the initial values. XRESET-signal is asynchronous to any external clock. The reset mode doubles as a full-powerdown mode, where both digital and analog parts of VS1053b are in minimum power consumption stage, and where clocks are stopped. Also XTALO is grounded.

When XRESET is asseted, all output pins go to their default states. All input pins will go to high-impedance state (to input state), except SO, which is still controlled by the XCS.

After a hardware reset (or at power-up) DREQ will stay down for around 22000 clock cycles, which means an approximate 1.8 ms delay if VS1053b is run at 12.288 MHz. After this the user should set such basic software registers as SCI_MODE, SCI_BASS, SCI_CLOCKF, and SCI_VOL before starting decoding. See section 9.6 for details.

If the input clock is 24..26 MHz, SM_CLK_RANGE should be set as soon as possible after a chip reset without waiting for DREQ.

Internal clock can be multiplied with a PLL. Supported multipliers through the SCI_CLOCKF register are $1.0 \times \ldots 5.0\times$ the input clock. Reset value for Internal Clock Multiplier is $1.0\times$. If typical values are wanted, the Internal Clock Multiplier needs to be set to $3.5\times$ after reset. Wait until DREQ rises, then write value 0x9800 to SCI_CLOCKF (register 3). See section 9.6.4 for details.

10.3 Software Reset

In some cases the decoder software has to be reset. This is done by activating bit SM_RESET in register SCI_MODE (Chapter 9.6.1). Then wait for at least 2 μs, then look at DREQ. DREQ

10 操作

10.1 时钟系统

VS1053b工作在单一主时钟基频下，标称值为12.288 MHz。
该时钟可由外部电路（连接至XTALI引脚）或内部时钟晶振接口（XTALI与XTALO引脚）产生，供模拟部件使用并决定最高可用采样率。使用12.288 MHz时钟时，最高支持48000赫兹的所有采样率。

当SCI_MODE寄存器的SM_CLK_RANGE位设为1时，VS1053b也可使用24~26 MHz时钟源。" // 注：寄存器名保留原文此时系统时钟在输入端口进行二分频，芯片实际获得12~13 MHz的输入时钟。

10.2 硬件复位

当XRESET信号被驱动至低电平时，VS1053b将执行复位操作，所有控制寄存器及内部状态均被设置为初始值。XRESET信号与任何外部时钟均保持异步关系。该复位模式兼具全断电模式功能：此时VS1053b的数字与模拟模块均处于最低功耗状态，时钟停止运行，且XTALO引脚接地。

当XRESET信号有效时，所有输出引脚将返回默认状态。除仍受XCS控制的SO引脚外，所有输入引脚将进入高阻态（输入状态）。

硬件复位（或上电）后，DREQ信号将保持低电平约22000个时钟周期——若VS1053b以12.288 MHz运行，则对应约1.8毫秒延迟。此后用户应在开始解码前设置基本软件寄存器，包括SCI_MODE、SCI_BASS、SCI_CLOCKF及SCI_VOL。 详情参见章节 9.6。

若输入时钟为24..26 MHz，应在芯片复位后立即设置SM_CLK_RANGE，无需等待DREQ信号。

内部时钟可通过锁相环(PLL)倍频。通过SCI_CLOCKF寄存器支持的倍频系数为 1.0倍...5.0倍输入时钟。内部时钟倍频器复位默认值为 1.0倍。 若需典型工作值，复位后应将内部时钟倍频器设置为 3.5倍。 等待DREQ信号变高，随后向SCI_CLOCKF（寄存器3）写入值0x9800。 详情参见章节 9.6.4。

10.3 软件复位

某些情况下需复位解码器软件。操作方法：置位SCI_MODE寄存器中的SM_RESET位（第9.6.1章），等待至少2微秒后检测DREQ信号。DREQ

will stay down for about 22000 clock cycles, which means an approximate 1.8 ms delay if VS1053b is run at 12.288 MHz. After DREQ is up, you may continue playback as usual.

As opposed to all earlier VS10XX chips, it is not recommended to do a software reset between songs. This way the user may be sure that even files with low samplerates or bitrates are played right to their end.

10.4 Low Power Mode

If you need to keep the system running while not decoding data, but need to lower the power consumption, you can use the following tricks.

- Select the $1.0\times$ clock by writing 0x0000 to SCI_CLOCKF. This disables the PLL and saves some power.
- Write a low non-zero value, such as 0x0010 to SCI_AUDATA. This will reduce the samplerate and the number of audio interrupts required. Between audio interrupts the VSDSP core will just wait for an interrupt, thus saving power.
- Turn off all audio post-processing (tone controls and EarSpeaker).
- If possible for the application, write 0xffff to SCI_VOL to disable the analog drivers.

To return from low-power mode, revert register values in reverse order.

Note: The low power mode consumes significantly more electricity than hardware reset.

10.5 Play and Decode

This is the normal operation mode of VS1053b. SDI data is decoded. Decoded samples are converted to analog domain by the internal DAC. If no decodable data is found, SCI_HDAT0 and SCI_HDAT1 are set to 0.

When there is no input for decoding, VS1053b goes into idle mode (lower power consumption than during decoding) and actively monitors the serial data input for valid data.

将保持低电平约22000个时钟周期，若VS1053b运行于12.288 MHz，则产生约1.8毫秒延迟。DREQ恢复高电平后，可正常继续播放。

与早期所有VS10xx系列芯片不同，不建议在歌曲播放间隙执行软件复位操作。通过这种方式，用户可确保即使是低采样率或比特率的文件也能完整播放至结束。

10.4 低功耗模式

若需在非解码状态下维持系统运行并降低功耗，可采用以下技巧：

- 向SCI_CLOCKF寄存器写入0x0000选择1.0×时钟。此举将禁用PLL以节省功耗。
- 向SCI_AUDATA写入较小的非零值（如0x0010），可降低采样率并减少所需音频中断次数。在音频中断的间隙，VSDSP核心将保持等待中断状态，从而降低功耗。
- 关闭所有音频后处理功能（音调控制及耳机/扬声器组合）。
- 若应用条件允许，请向SCI_VOL寄存器写入0xffff以禁用模拟驱动器。

退出低功耗模式时，按逆序恢复寄存器原始值。

注意：低功耗模式的耗电量显著高于硬件复位状态。

10.5 播放与解码

此乃VS1053b标准工作模式：对SDI数据进行解码处理。解码后的采样数据通过内置DAC转换为模拟信号。若未检测到可解码数据，则SCI_HDAT0与SCI_HDAT1寄存器将置零。

当无待解码输入时，VS1053b进入空闲模式（功耗低于解码状态），并持续监测串行数据输入端口以捕获有效数据。

10.5.1 Playing a Whole File

This is the default playback mode.

1. Send an audio file to VS1053b.
2. Read extra parameter value endFillByte (Chapter 10.11).
3. Send at least 2052 bytes of endFillByte[7:0].
4. Set SCI_MODE bit SM_CANCEL.
5. Send at least 32 bytes of endFillByte[7:0].
6. Read SCI_MODE. If SM_CANCEL is still set, go to 5. If SM_CANCEL hasn't cleared after sending 2048 bytes, do a software reset (this should be extremely rare).
7. The song has now been successfully sent. HDAT0 and HDAT1 should now both contain 0 to indicate that no format is being decoded. Return to 1.

10.5.2 Cancelling Playback

Cancelling playback of a song is a normal operation when the user wants to jump to another song while doing playback.

1. Send a portion of an audio file to VS1053b.
2. Set SCI_MODE bit SM_CANCEL.
3. Continue sending audio file, but check SM_CANCEL after every 32 bytes of data. If it is still set, goto 3. If SM_CANCEL doesn't clear after 2048 bytes or one second, do a software reset (this should be extremely rare).
4. When SM_CANCEL has cleared, read extra parameter value endFillByte (Chapter 10.11).
5. Send 2052 bytes of endFillByte[7:0].
6. HDAT0 and HDAT1 should now both contain 0 to indicate that no format is being decoded. You can now send the next audio file.

10.5.3 Fast Play

VS1053b allows fast audio playback. If your microcontroller can feed data fast enough to the VS1053b, this is the preferred way to fast forward audio.

1. Start sending an audio file to VS1053b.
2. To set fast play, set extra parameter value playSpeed (Chapter 10.11).
3. Continue sending audio file.
4. To exit fast play mode, write 1 to playSpeed.

To estimate whether or not your microcontroller can feed enough data to VS1053b in fast play mode, see contents of extra parameter value byteRate (Chapter 10.11). Note that byteRate contains the data speed of the file played back at nominal speed even when fast play is active.

Note: Play speed is not reset when song is changed.

10.5.1 完整文件播放

此模式为默认播放模式。

1. 将音频文件传输至VS1053b芯片。
2. 读取额外参数值endFillByte（第10.11章）。
3. 发送至少2052字节的endFillByte[7:0]。
4. 设置SCI_MODE寄存器的SM_CANCEL位。
5. 发送至少32字节的endFillByte[7:0]。
6. 读取SCI_MODE寄存器。若SM_CANCEL位仍被置位，则跳转至步骤5。 若发送2048字节后SM_CANCEL仍未清除，则执行软件复位（此情况应极为罕见）。
7. 歌曲数据现已成功发送完毕。此时HDAT0与HDAT1寄存器值均应为0，表示无格式正在解码。返回步骤1。

10.5.2 取消播放

当用户需要在播放过程中跳转至另一首歌曲时，取消当前歌曲播放属于常规操作。

1. 向VS1053b发送部分音频文件数据。
2. 设置SCI_MODE寄存器的SM_CANCEL位。
3. 继续发送音频文件数据，但每发送32字节数据后需检测SM_CANCEL状态。若该标志仍被置位，则转到步骤3。 若SM_CANCEL标志在传输2048字节或1秒后仍未清除，则执行软件复位（此情况应极为罕见）。
4. 当SM_CANCEL标志清除后，读取额外参数值endFillByte（参见章节10.11）。
5. 发送2052字节的endFillByte[7:0]数据。
6. 此时HDAT0和HDAT1寄存器值均应显示为0，表明当前无解码格式。
   此时可发送下一个音频文件。

10.5.3 快速播放

VS1053b支持音频快速播放功能。若微控制器能向VS1053b提供足够快的数据流，此为实现音频快进的首选方法。

1. 开始向VS1053b发送音频文件。
2. 设置额外参数playSpeed以启用快速播放（参见章节10.11）。
3. 继续发送音频文件数据流。
4. 向playSpeed参数写入1可退出快速播放模式。

评估微控制器能否在快速播放模式下向VS1053b提供足够数据时，请查阅额外参数值byteRate（第10.11章）。注意：即使在快速播放激活时，byteRate仍包含标称速度下的文件数据速率。

注意：切换歌曲时播放速度不会重置。

10.5.4 Fast Forward and Rewind without Audio

To do fast forward and rewind you need the capability to do random access to the audio file. Unfortunately fast forward and rewind isn't available at all times, like when file headers are being read.

1. Send a portion of an audio file to VS1053b.
2. When random access is required, read SCI_STATUS bit SS_DO_NOT_JUMP. If that bit is set, random access cannot be performed, so go back to 1.
3. Read extra parameter value endFillByte (Chapter 10.11).
4. Send at least 2048 bytes of endFillByte[7:0].
5. Jump forwards or backwards in the file.
6. Continue sending the file.

Note: It is recommended that playback volume is decreased by e.g. 10 dB when fast forwarding/rewinding.

Note: Register DECODE_TIME does not take jumps into account.

Note: Midi is not suitable for random-access. You can implement fast forward using the `playSpeed` extra parameter to select 1-128× play speed. SCI_DECODE_TIME also speeds up. If necessary, rewind can be implemented by restarting decoding of a MIDI file and fast playing to the appropriate place. SCI_DECODE_TIME can be used to decide when the right place has been reached.

10.5.5 Maintaining Correct Decode Time

When fast forward and rewind operations are performed, there is no way to maintain correct decode time for most files. However, WMA and Ogg Vorbis files offer exact time information in the file. To use accurate time information whenever possible, use the following algorithm:

1. Start sending an audio file to VS1053b.
2. Read extra parameter value pair positionMsec (Chapter 10.11).
3. If positionMsec is -1, show you estimation of decoding time using DECODE_TIME (and your estimate of file position if you have performed fast forward / rewind operations).
4. If positionMsec is not -1, use this time to show the exact position in the file.

10.5.4 无音频快速前进与倒带

实现快速前进/倒带功能需具备对音频文件的随机访问能力。
需注意快速前进/倒带功能并非始终可用，例如在读取文件头期间。

1. 向VS1053b发送部分音频文件数据。
2. 当需要随机访问时，读取SCI_STATUS寄存器的SS_DO_NOT_JUMP位。若该位置位，则无法执行随机访问，此时需返回步骤1。
3. 读取额外参数值endFillByte（第10.11章）。
4. 发送至少2048字节的endFillByte[7:0]填充数据。
5. 在文件中执行前进或后退跳转操作。
6. 继续发送音频文件数据。

注意：建议在快速前进/倒带时将播放音量降低10 dB。

注意：寄存器 DECODE_TIME 不考虑时间跳跃。

注意：MIDI 格式不适合随机访问播放。可通过 playSpeedextra 参数选择 1-128×播放速度来实现快进功能。SCI_DECODE_TIME 数值也会加速变化。必要时可通过重启 MIDI 文件解码并快速播放至目标位置来实现回退操作。SCI_DECODE_TIME 可用于判定是否到达目标位置。

10.5.5 保持正确解码时间

执行快进与回退操作时，多数文件无法维持正确的解码时间记录。但 WMA 和 Ogg Vorbis 格式的文件中包含精确时间信息。为尽可能使用精确时间信息，请采用以下算法：

1. 开始向VS1053b发送音频文件。
2. 读取参数值对 positionMsec（第 10.11 章）。
3. 若 positionMsec 为 -1，则使用 DECODE_TIME 值（若执行过快进/回退操作，需结合文件位置估算）显示解码时间预估值。
4. 若positionMsec不为-1，则使用该时间值显示文件中的精确位置。

10.6 Feeding PCM data

VS1053b can be used as a PCM decoder by sending a WAV file header. If the length sent in the WAV header is 0xFFFFFFFF, VS1053b will stay in PCM mode indefinitely (or until SM_CANCEL has been set). 8-bit linear and 16-bit linear audio is supported in mono or stereo. A WAV header looks like this:

File Offset	Field Name	Size	Bytes	Description
0	ChunkID	4	"RIFF"	
4	ChunkSize	4	0xff 0xff 0xff 0xff	
8	Format	4	"WAVE"	
12	SubChunk1ID	4	"fmt "	
16	SubChunk1Size	4	0x10 0x0 0x0 0x0	16
20	AudioFormat	2	0x1 0x0	Linear PCM
22	NumOfChannels	2	C0 C1	1 for mono, 2 for stereo
24	SampleRate	4	S0 S1 S2 S3	0x1f40 for 8 kHz
28	ByteRate	4	R0 R1 R2 R3	0x3e80 for 8 kHz 16-bit mono
32	BlockAlign	2	A0 A1	0x02 0x00 for mono, 0x04 0x00 for stereo 16-bit
34	BitsPerSample	2	B0 B1	0x10 0x00 for 16-bit data
52	SubChunk2ID	4	"data"	
56	SubChunk2Size	4	0xff 0xff 0xff 0xff	Data size

The rules to calculate the four variables are as follows:

- $S =$ sample rate in Hz, e.g. 44100 for 44.1 kHz.
- For 8-bit data $B = 8$, and for 16-bit data $B = 16$.
- For mono data $C = 1$, for stereo data $C = 2$.
- $A = \frac{C \times B}{8}$.
- $R = S \times A$.

Example: A 44100 Hz 16-bit stereo PCM header would read as follows:

```
0000  52 49 46 46 ff ff ff ff  57 41 56 45 66 6d 74 20  |RIFF....WAVEfmt |
0100  10 00 00 00 01 00 02 00  44 ac 00 00 10 b1 02 00  |........D.......|
0200  04 00 10 00 64 61 74 61  ff ff ff ff              |....data....|
```

10.7 Ogg Vorbis Recording

Ogg Vorbis is an open file format that allows for very high sound quality with low to medium bitrates.

Ogg Vorbis recording is activated by loading the Ogg Vorbis Encoder Application to the 16 KiB program RAM memory of the VS1053b. After activation, encoder results can be read from registers SCI_HDAT0 and SCI_HDAT1, much like when using PCM/ADPCM recording (Chapter 10.8).

Three profiles are provided: one for high-quality stereo recording at a bitrate of approx. 140 kbit/s, and two for speech-quality mono recording at a bitrates between 15 and 30 kbit/s.

To use the Ogg Vorbis Encoder application, please load the application from VLSI Solution's Web page *http://www.vlsi.fi/en/support/software/vs10xxapplications.html* and read the accompanying documentation.

10.6 PCM数据输入

通过发送WAV文件头，可将VS1053b用作PCM解码器。若WAV文件头中发送的长度值为0xFFFFFFFFffffffff，VS1053b将无限期保持PCM模式（直至设置SM_CANCEL寄存器）。支持单声道/立体声的8位线性和16位线性音频。WAV文件头结构如下：

文件偏移量	字段名称	大小	字节	描述
0	区块标识	4	"RIFF"	
4	数据块大小	4	0xff 0xff 0xff 0xff	
8	格式	4	"WAVE"	
12	子块1标识	4	"fmt "	
16	子块1大小	4	0x10 0x0 0x0 0x0	16
20	音频格式	2	0x1 0x0	线性PCM
22	声道数	2	C0 C1	1代表单声道，2代表立体声
24	采样率	4	S0 S1 S2 S3	0x1f40对应8kHz采样率" // 数值与单位说明分离
28	字节速率	4	R0 R1 R2 R3	0x3e80对应8kHz 16位单声道" // 数值与参数说明分离
32	块对齐	2	A0 A1	0x02 0x00代表单声道，0x04 0x00代表16位立体声" // 数值与参数说明分离
34	每样本位数	2	B0 B1	0x10 0x00 代表16位数据
52	子区块2标识符	4	"数据"	
56	子区块2大小	4	0xff 0xff 0xff 0xff	数据大小

四个变量的计算规则如下：

- S = 采样率（单位：赫兹），例如44100代表44.1 kHz。
- 8位数据对应 $B = 8$，16位数据对应 $B = 16$。
- 单声道数据 $C = 1$，立体声数据 $C = 2$。
- $A = \frac{C \times B}{8}$.
- $R = S \times A$ 。

示例：44100赫兹16位立体声PCM文件头如下所示：

```
0000 52 49 46 46 ff ff ff ff 57 41 56 45 66 6d 74 20 |RIFF....WAVEfmt |
0100 10 00 00 00 01 00 02 00 44 ac 00 00 10 b1 02 00 |........................................D.......|
0200 04 00 10 00 64 61 74 61 ff ff ff ff                                        |....data....|
```

10.7 Ogg Vorbis 录制

Ogg Vorbis 是一种开放文件格式，可在中低比特率下实现极高的音质。

通过将 Ogg Vorbis 编码器应用程序加载至 VS1053b 的 16 千字节程序 RAM 存储器来激活录制功能。激活后，可从 SCI_HDAT0 和 SCI_HDAT1 寄存器读取编码结果，操作方式类似于 PCM /ADPCM 录制（参见章节 10.8）。

提供三种配置文件：一种支持约 140 千比特/秒的高质量立体声录制，另两种支持 15 至 30 千比特/比特率的语音质量单声道录制。

使用 Ogg Vorbis 编码器应用程序时，请从 VLSI Solution 官网加载程序*http://www.vlsi.fi/en/support/software/vs10xxapplications.html*并阅读随附文档。

10.8 PCM/ADPCM Recording

This chapter explains how to create RIFF/WAV file in PCM or IMA ADPCM format. IME ADPCM is a widely supported ADPCM format and many PC audio playback programs can play it. IMA ADPCM recording gives roughly a compression ratio of 4:1 compared to linear, 16-bit audio. This makes it possible to record for example ono 8 kHz audio at 32.44 kbit/s.

VS1053 has a stereo ADC, thus also two-channel (separate AGC, if AGC enabled) and stereo (common AGC, if AGC enabled) modes are available. Mono recording mode selects either left or right channel. Left channel is either MIC or LINE1 depending on the SCI_MODE register.

10.8.1 Activating ADPCM Mode

Register	Bits	Description
SCI_MODE	2, 12, 14	Start ADPCM mode, select MIC/LINE1
SCI_AICTRL0	15..0	Sample rate 8000..48000 Hz (read at recording startup)
SCI_AICTRL1	15..0	Recording gain (1024 = 1×) or 0 for automatic gain control
SCI_AICTRL2	15..0	Maximum autogain amplification (1024 = 1×, 65535 = 64×)
SCI_AICTRL3	1..0	0 = joint stereo (common AGC), 1 = dual channel (separate AGC), 2 = left channel, 3 = right channel
	2	0 = IMA ADPCM mode, 1 = LINEAR PCM mode
	15..3	reserved, set to 0

PCM / IMA ADPCM recording mode is activated by setting bits SM_RESET and SM_ADPCM in SCI_MODE. Line input 1 is used instead of differential mic input if SM_LINE1 is set. Before activating ADPCM recording, user **must** write the right values to SCI_AICTRL0 and SCI_AICTRL3. These values are only read at recording startup. SCI_AICTRL1 and SCI_AICTRL2 can be altered anytime, but it is preferable to write good init values before activation.

SCI_AICTRL1 controls linear recording gain. 1024 is equal to digital gain 1, 512 is equal to digital gain 0.5 and so on. If the user wants to use automatic gain control (AGC), SCI_AICTRL1 should be set to 0. Typical speech applications usually are better off using AGC, as this takes care of relatively uniform speech loudness in recordings.

SCI_AICTRL2 controls the maximum AGC gain. This can be used to limit the amplification of noise when there is no signal. If SCI_AICTRL2 is zero, the maximum gain is initialized to 65535 (64×), i.e. whole range is used.

For example:

```
WriteVS10xxRegister(SCI_AICTRL0, 16000U);
WriteVS10xxRegister(SCI_AICTRL1, 0);
WriteVS10xxRegister(SCI_AICTRL2, 4096U);
WriteVS10xxRegister(SCI_AICTRL3, 0);
WriteVS10xxRegister(SCI_MODE, ReadVS10xxRegister(SCI_MODE) |
                              SM_RESET | SM_ADPCM | SM_LINE1);
WriteVS10xxPatch(); /* Only for VS1053b and VS8053b */
```

selects 16 kHz, stereo mode with automatic gain control and maximum amplification of $4\times$.

10.8 PCM/ADPCM 录音功能

本章阐述如何创建PCM或IMA ADPCM格式的RIFF/WAV文件。IMA ADPCM是一种广泛支持的ADPCM格式，多数PC音频播放程序均可解码。相较于16位线性音频，IMA ADPCM录音可提供约4:1的压缩比。
这使得录制8kHz音频（例如32.44 kbit/s码率）成为可能。

VS1053配备立体声ADC，因此支持双声道模式（若启用AGC则为独立AGC）和立体声模式（若启用AGC则为共用AGC）。 单声道录音模式可选择左或右声道。左声道信号源（麦克风或LINE1）由SCI_MODE寄存器配置决定。

10.8.1 激活ADPCM模式

寄存器	位	描述
SCI_MODE	2, 12, 14	启动ADPCM模式，选择麦克风/LINE1输入
SCI_AICTRL0	15..0	采样率范围8000..48000赫兹（在录制启动时读取）
SCI_AICTRL1	15..0	录音增益（1024 = 1×倍），若为0则启用自动增益控制
SCI_AICTRL2	15..0	最大自动增益放大倍数（1024 = 1×倍，65535 = 64×倍）
SCI_AICTRL3	1..0 2 15..3	0 = 联合立体声（共用AGC），1 = 双声道（独立AGC）， 2 = 左声道，3 = 右声道 0 = IMA ADPCM模式，1 = 线性PCM模式 保留位，需设置为0

通过设置SCI_MODE寄存器中的SM_RESET和SM_ADPCM位激活PCM/IMA ADPCM录音模式。若设置SM_LINE1则使用线路输入1替代差分麦克风输入。激活ADPCM录音前，用户必须向SCI_AICTRL0和SCI_AICTRL3写入正确数值。
这些数值仅在录音启动时读取。SCI_AICTRL1和SCI_AICTRL2可随时修改，但建议在激活前写入合适的初始值。

SCI_AICTRL1寄存器控制线性录音增益。1024对应数字增益1，512对应数字增益0.5，以此类推。若需启用自动增益控制(AGC)，应将SCI_AICTRL1设置为0。 典型语音应用通常更适合采用AGC，因其可确保录音中语音响度保持相对一致。

SCI_AICTRL2寄存器控制AGC最大增益值。该功能可限制无信号时的噪声放大。若SCI_AICTRL2为零，则最大增益初始化为65535(64×)，即使用整个增益范围。

例如：

```
WriteVS10xxRegister(SCI_AICTRL0, 16000U);
WriteVS10xxRegister(SCI_AICTRL1, 0);
WriteVS10xxRegister(SCI_AICTRL2, 4096U);
写入VS10xx寄存器(SCI_AICTRL3, 0);
写入VS10xx寄存器(SCI_MODE, 读取VS10xx寄存器(SCI_MODE) | SM_R
                                    ESET | SM_ADPCM | SM_LINE1);
写入VS10xx补丁(); /* 仅适用于VS1053b和VS8053b */
```

选择16千赫兹立体声模式，带自动增益控制和最大 4× 的放大倍数

`WriteVS10xxPatch()` should perform the following SCI writes (only for VS1053b and VS8053b):

Register	Reg. No	Value
SCI_WRAMADDR	0x7	0x8010
SCI_WRAM	0x6	0x3e12
SCI_WRAM	0x6	0xb817
SCI_WRAM	0x6	0x3e14
SCI_WRAM	0x6	0xf812
SCI_WRAM	0x6	0x3e01
SCI_WRAM	0x6	0xb811
SCI_WRAM	0x6	0x0007
SCI_WRAM	0x6	0x9717
SCI_WRAM	0x6	0x0020
SCI_WRAM	0x6	0xffd2
SCI_WRAM	0x6	0x0030
SCI_WRAM	0x6	0x11d1
SCI_WRAM	0x6	0x3111
SCI_WRAM	0x6	0x8024
SCI_WRAM	0x6	0x3704
SCI_WRAM	0x6	0xc024
SCI_WRAM	0x6	0x3b81
SCI_WRAM	0x6	0x8024
SCI_WRAM	0x6	0x3101
SCI_WRAM	0x6	0x8024
SCI_WRAM	0x6	0x3b81
SCI_WRAM	0x6	0x8024
SCI_WRAM	0x6	0x3f04
SCI_WRAM	0x6	0xc024
SCI_WRAM	0x6	0x2808
SCI_WRAM	0x6	0x4800
SCI_WRAM	0x6	0x36f1
SCI_WRAM	0x6	0x9811
SCI_WRAMADDR	0x7	0x8028
SCI_WRAM	0x6	0x2a00
SCI_WRAM	0x6	0x040e

This patch is also available from VLSI Solution's web page *http://www.vlsi.fi/en/support/software/vs10xxpatches.html* by the name of *VS1053b IMA ADPCM Encoder Fix*, and it is also part of the VS1053b patches package.

10.8.2 Reading PCM / IMA ADPCM Data

After PCM / IMA ADPCM recording has been activated, registers SCI_HDAT0 and SCI_HDAT1 have new functions.

The PCM / IMA ADPCM sample buffer is 1024 16-bit words. The fill status of the buffer can be read from SCI_HDAT1. If SCI_HDAT1 is greater than 0, you can read as many 16-bit words from SCI_HDAT0. If the data is not read fast enough, the buffer overflows and returns to empty state.

Note: if SCI_HDAT1 ≥ 768, it may be better to wait for the buffer to overflow and clear before reading samples. That way you may avoid buffer aliasing.

In IMA ADPCM mode each mono IMA ADPCM block is 128 words, i.e. 256 bytes, and stereo

写入VS10xx补丁()应执行以下SCI写入操作（仅适用于VS1053b和VS8053b）：

寄存器	寄存器编号	取值
SCI_WRAMADDR	0x7	0x8010
SCI_WRAM	0x6	0x3e12
SCI_WRAM	0x6	0xb817
SCI_WRAM	0x6	0x3e14
SCI_WRAM	0x6	0xf812
SCI_WRAM	0x6	0x3e01
SCI_WRAM	0x6	0xb811
SCI_WRAM	0x6	0x0007
SCI_WRAM	0x6	0x9717
SCI_WRAM	0x6	0x0020
SCI_WRAM	0x6	0xffd2
SCI_WRAM	0x6	0x0030
SCI_WRAM	0x6	0x11d1
SCI_WRAM	0x6	0x3111
SCI_WRAM	0x6	0x8024
SCI_WRAM	0x6	0x3704
SCI_WRAM	0x6	0xc024
SCI_WRAM	0x6	0x3b81
SCI_WRAM	0x6	0x8024
SCI_WRAM	0x6	0x3101
SCI_WRAM	0x6	0x8024
SCI_WRAM	0x6	0x3b81
SCI_WRAM	0x6	0x8024
SCI_WRAM	0x6	0x3f04
SCI_WRAM	0x6	0xc024
SCI_WRAM	0x6	0x2808
SCI_WRAM	0x6	0x4800
SCI_WRAM	0x6	0x36f1
SCI_WRAM	0x6	0x9811
SCI_WRAMADDR	0x7	0x8028
SCI_WRAM	0x6	0x2a00
SCI_WRAM	0x6	0x040e

该补丁同样可从VLSI Solution官网获取
*http://www.vlsi.fi/en/support/software/vs10xxpatches.html*文件名为*VS1053b IMA ADPCM*编码器修复补丁，该补丁也包含在VS1053b补丁包中。

10.8.2 读取PCM/IMA ADPCM数据

激活PCM/IMA ADPCM录音功能后，寄存器SCI_HDAT0和SCI_HDAT1将具有新功能。

PCM/IMA ADPCM采样缓冲区容量为1024个16位字。可通过SCI_HDAT1读取缓冲区填充状态。若SCI_HDAT1值大于0，则可从SCI_HDAT0读取相应数量的16位字。若数据读取速度不足，缓冲区将溢出并返回空状态。

注意：当SCI_HDAT1 ≥ 768时，建议等待缓冲区溢出清空后再读取采样数据。此操作可避免缓冲区混叠现象。

IMA ADPCM模式下，每个单声道IMA ADPCM数据块为128字（即256字节），而立体声

IMA ADPCM block is 256 words, i.e. 512 bytes. If you wish to interrupt reading data and possibly continue later, please stop at the boundary. This way whole blocks are skipped and the encoded stream stays valid.

10.8.3 Adding a PCM RIFF Header

To make your PCM file a RIFF / WAV file, you have to add a header to the data. The following shows a header for a mono file. Note that 2- and 4-byte values are little-endian (lowest byte first).

File Offset	Field Name	Size	Bytes	Description
0	ChunkID	4	"RIFF"	
4	ChunkSize	4	F0 F1 F2 F3	File size - 8
8	Format	4	"WAVE"	
12	SubChunk1ID	4	"fmt "	
16	SubChunk1Size	4	0x10 0x0 0x0 0x0	20
20	AudioFormat	2	0x01 0x0	0x1 for PCM
22	NumOfChannels	2	C0 C1	1 for mono, 2 for stereo
24	SampleRate	4	R0 R1 R2 R3	0x1f40 for 8 kHz
28	ByteRate	4	B0 B1 B2 B3	0x3e80 for 8 kHz mono
32	BlockAlign	2	0x02 0x00	2 for mono, 4 for stereo
34	BitsPerSample	2	0x10 0x00	16 bits / sample
36	SubChunk3ID	4	"data"	
40	SubChunk3Size	4	D0 D1 D2 D3	Data size (File Size-36)
44	Samples...			Audio samples

The values in the table are calculated as follows:
$R = F_s$ (see Chapter 10.8.1 to see how to calculate F_s)
$B = 2 \times F_s \times C$

If you know beforehand how much you are going to record, you may fill in the complete header before any actual data. However, if you don't know how much you are going to record, you have to fill in the header size datas F and D after finishing recording.

The PCM data is read from SCI_HDAT0 and written into file as follows. The high 8 bits of SCI_HDAT0 should be written as the first byte to a file, then the low 8 bits. Note that this is contrary to the default operation of some 16-bit microcontrollers, and you may have to take extra care to do this right.

Below is an example of a valid header for a 44.1 kHz mono PCM file that has a final length of 1798768 (0x1B7270) bytes:

```
0000  52 49 46 46 68 72 1b 00  57 41 56 45 66 6d 74 20  |RIFFhr..WAVEfmt |
0010  10 00 00 00 01 00 01 00  80 bb 00 00 00 77 01 00  |.............w..|
0020  02 00 10 00 64 61 74 61  44 72 1b 00              |....dataDr......|
```

IMA ADPCM数据块为256字（即512字节）。 若需中断数据读取并可能后续继续，请在边界处停止操作。此方式可跳过完整数据块，同时保持编码流有效性。

10.8.3 添加PCM RIFF头

将PCM文件转为RIFF/WAV音频格式需添加文件头。以下展示单声道文件的文件头结构。注意：2字节与4字节数值采用小端序（最低有效字节在前）。

文件偏移量	字段名称	大小	字节	描述
0	区块标识	4	"RIFF"	
4	数据块大小	4	F0 F1 F2 F3	文件大小 - 8
8	格式	4	"WAVE"	
12	子块1标识	4	"fmt "	
16	子块1大小	4	0x10 0x0 0x0 0x0	20
20	音频格式	2	0x01 0x0	0x01 表示PCM格式
22	声道数	2	C0 C1	1代表单声道，2代表立体声
24	采样率	4	R0 R1 R2 R3	0x1f40对应8kHz采样率" // 数值与单位说明分离
28	字节速率	4	B0 B1 B2 B3	0x3e80 对应8千赫兹单声道
32	块对齐	2	0x02 0x00	2表示单声道，4表示立体声
34	每样本位数	2	0x10 0x00	16位/采样
36	SubChunk3ID	4	"数据"	
40	SubChunk3Size	4	D0 D1 D2 D3	数据大小（文件大小-36）
44	采样...			音频采样

表中数值按以下方式计算：
$R = F_s$ （计算 F_s的方法参见第10.8.1章）
$B = 2 \times F_s \times C$

若预先确定录制数据量，可在写入实际数据前填写完整文件头信息但若录制数据量未知，则需在录制完成后填写文件头尺寸数据 F与 D

PCM数据按以下方式从SCI_HDAT0读取并写入文件：应将SCI_HDAT0的高8位作为首字节写入文件，再写入低8位注意：此操作与某些16位微控制器的默认方式相反，需特别确保操作正确性

以下是一个有效的44.1千赫兹单声道PCM文件头示例，其最终长度为1798768（0x1B7270）字节：

```
0000 52 49 46 46 68 72 1b 00 57 41 56 45 66 6d 74 20 |RIFFhr..WAVEfmt |
0010 10 00 00 00 01 00 01 00 80 bb 00 00 00 77 01 00 |.............................w..|
0020 02 00 10 00 64 61 74 61 44 72 1b 00                         |....dataDr.....|
```

10.8.4 Adding an IMA ADPCM RIFF Header

To make your IMA ADPCM file a RIFF / WAV file, you have to add a header to the data. The following shows a header for a mono file. Note that 2- and 4-byte values are little-endian (lowest byte first).

File Offset	Field Name	Size	Bytes	Description
0	ChunkID	4	`"RIFF"`	
4	ChunkSize	4	F0 F1 F2 F3	File size - 8
8	Format	4	`"WAVE"`	
12	SubChunk1ID	4	`"fmt "`	
16	SubChunk1Size	4	0x14 0x0 0x0 0x0	20
20	AudioFormat	2	0x11 0x0	0x11 for IMA ADPCM
22	NumOfChannels	2	C0 C1	1 for mono, 2 for stereo
24	SampleRate	4	R0 R1 R2 R3	0x1f40 for 8 kHz
28	ByteRate	4	B0 B1 B2 B3	0xfd7 for 8 kHz mono
32	BlockAlign	2	0x00 0x01	256 for mono, 512 for stereo
34	BitsPerSample	2	0x04 0x00	4-bit ADPCM
36	ByteExtraData	2	0x02 0x00	2
38	ExtraData	2	0xf9 0x01	Samples per block (505)
40	SubChunk2ID	4	`"fact"`	
44	SubChunk2Size	4	0x4 0x0 0x0 0x0	4
48	NumOfSamples	4	S0 S1 S2 S3	
52	SubChunk3ID	4	`"data"`	
56	SubChunk3Size	4	D0 D1 D2 D3	Data size (File Size-60)
60	Block1	256		First ADPCM block, 512 bytes for stereo
316	...			More ADPCM data blocks

If we have n audio blocks, the values in the table are as follows:
$F = n \times C \times 256 + 52$
$R = F_s$ (see Chapter 10.8.1 to see how to calculate F_s)
$B = \frac{F_s \times C \times 256}{505}$
$S = n \times 505$. $D = n \times C \times 256$

If you know beforehand how much you are going to record, you may fill in the complete header before any actual data. However, if you don't know how much you are going to record, you have to fill in the header size datas F, S and D after finishing recording.

The 128 words (256 words for stereo) of an ADPCM block are read from SCI_HDAT0 and written into file as follows. The high 8 bits of SCI_HDAT0 should be written as the first byte to a file, then the low 8 bits. Note that this is contrary to the default operation of some 16-bit microcontrollers, and you may have to take extra care to do this right.

To see if you have written the mono file in the right way check bytes 2 and 3 (the first byte counts as byte 0) of each 256-byte block. Byte 2 should be $0..88$ and byte 3 should be zero. For stereo you check bytes 2, 3, 6, and 7 of each 512-byte block. Bytes 2 and 6 should be $0..88$. Bytes 3 and 7 should be zero.

Below is an example of a valid header for a 44.1 kHz stereo IMA ADPCM file that has a final length of 10038844 (0x992E3C) bytes:

10.8.4 添加IMA ADPCM RIFF文件头

要将IMA ADPCM文件转换为RIFF/WAV格式，需在数据前添加文件头。以下展示单声道文件的文件头结构。注意：2字节与4字节数值采用小端序（最低有效字节在前）。

文件偏移量	字段名称	大小	字节	描述
0	区块标识	4	"RIFF"	
4	数据块大小	4	F0 F1 F2 F3	文件大小 - 8
8	格式	4	"WAVE"	
12	子块1标识	4	"fmt "	
16	子块1大小	4	0x14 0x0 0x0 0x0	20
20	音频格式	2	0x11 0x0	0x11（用于IMA ADPCM）
22	声道数	2	C0 C1	1代表单声道，2代表立体声
24	采样率	4	R0 R1 R2 R3	0x1f40对应8kHz采样率" // 数值与单位说明分离
28	字节速率	4	B0 B1 B2 B3	0xfd7（适用于8 kHz单声道）
32	块对齐	2	0x00 0x01	单声道为256，立体声为512
34	每样本位数	2	0x04 0x00	4位ADPCM
36	字节额外数据	2	0x02 0x00	2
38	额外数据	2	0xf9 0x01	每块采样数(505)
40	子区块2标识符	4	"fact"	
44	子区块2大小	4	0x4 0x0 0x0 0x0	4
48	采样总数	4	S0 S1 S2 S3	
52	SubChunk3ID	4	"数据"	
56	SubChunk3Size	4	D0 D1 D2 D3	数据大小（文件大小-60）
60	数据块1	256		首个ADPCM数据块，立体声时为512字节
316	...			更多ADPCM数据块

若有 n个音频块，则表中数值如下：
$F = n \times C \times 256 + 52$
$R = F_s$ （计算 F_s的方法参见第10.8.1章）
$B = \frac{F_s \times C \times 256}{505}$
$S = n \times 505$. $D = n \times C \times 256$

若预先确定录制数据量，可在写入实际数据前填写完整文件头信息但若预录数据量未知，则需在录制完成后填充头部大小数据 F、 S与 D。

ADPCM数据块的128字（立体声为256字）从SCI_HDAT0读取后按以下方式写入文件。应将SCI_HDAT0的高8位作为首字节写入文件，再写入低8位注意：此操作与某些16位微控制器的默认方式相反，需特别确保操作正确性

验证单声道文件正确性时，需检查每个256字节块的第2-3字节（首字节计为字节0）：第2字节应为 0至88，第3字节应为零。立体声模式下需检查每个512字节块的第2、3、6及7字节。第2和第6字节应为 0至88。
第3和第7字节应为零值。

以下是一个44.1千赫兹立体声IMA ADPCM文件的有效头部示例，其最终长度为10038844（0x992E3C）字节：

```
0000  52 49 46 46 34 2e 99 00  57 41 56 45 66 6d 74 20  |RIFF4...WAVEfmt |
0010  14 00 00 00 11 00 02 00  44 ac 00 00 a7 ae 00 00  |........D.......|
0020  00 02 04 00 02 00 f9 01  66 61 63 74 04 00 00 00  |........fact....|
0030  14 15 97 00 64 61 74 61  00 2e 99 00              |....data....|
```

10.8.5 Playing ADPCM Data

In order to play back your PCM / IMA ADPCM recordings, you have to have a file with a header as described in Chapter 10.8.3 or Chapter 10.8.4. If this is the case, all you need to do is to provide the ADPCM file through SDI as you would with any audio file.

10.8.6 Sample Rate Considerations

VS10xx chips that support IMA ADPCM playback are capable of playing back ADPCM files with any sample rate. However, some other programs may expect IMA ADPCM files to have some exact sample rates, like 8000 or 11025 Hz. Also, some programs or systems do not support sample rates below 8000 Hz.

If you want better quality with the expense of increased data rate, you can use higher sample rates, for example 16 kHz.

10.8.7 Record Monitoring Volume

In VS1053b writing to the SCI_VOL register during IMA ADPCM encoding does not change the volume. You need to set a suitable volume before activating the IMA ADPCM mode, or you can use the VS1053 hardware volume control register DAC_VOL directly.

For example:

```
WriteVS10xxRegister(SCI_WRAMADDR, 0xc045); /*DAC_VOL*/
WriteVS10xxRegister(SCI_WRAM,     0x0101);     /*-6.0 dB*/
```

The hardware volume control DAC_VOL (address 0xc045) allows 0.5 dB steps for both left (high 8 bits) and right channel (low 8 bits). The low 4 bits of both 8-bit values set the attenuation in 6 dB steps, the high 4 bits in 0.5 dB steps.

```
0000 52 49 46 46 34 2e 99 00 57 41 56 45 66 6d 74 20 |RIFF4...WAVEfmt |
0010 14 00 00 00 11 00 02 00 44 ac 00 00 a7 ae 00 00 |........................................D.......|
0020 00 02 04 00 02 00 f9 01 66 61 63 74 04 00 00 00 |........................................fact....|
0030 14 15 97 00 64 61 74 61 00 2e 99 00                                        |....data....|
```

10.8.5 播放ADPCM数据

要播放您的PCM/IMA ADPCM录音，必须准备包含第10.8.3章或第10.8.4章所述文件头的文件。满足此条件后，只需像处理普通音频文件一样通过SDI接口提供ADPCM文件即可实现播放。

10.8.6 采样率注意事项

支持IMA ADPCM播放的VS10xx系列芯片能够播放任意采样率的ADPCM文件。但某些其他程序可能要求IMA ADPCM文件具有特定采样率（如8000或11025赫兹）。此外，部分程序或系统不支持低于8000赫兹的采样率。

若追求更高音质且可接受数据速率提升的代价，可采用更高采样率，例如16千赫兹（kHz）。

10.8.7 录音监听音量

在VS1053b进行IMA ADPCM编码期间，向SCI_VOL寄存器写入数据不会改变音量。需在激活IMA ADPCM模式前设定合适音量，或直接使用VS1053硬件音量控制寄存器DAC_VOL。

例如：

```
WriteVS10xxRegister(SCI_WRAMADDR, 0xc045); /*DAC_VOL*/
WriteVS10xxRegister(SCI_WRAM, 0x0101); /*-6.0 dB*/
```

硬件音量控制器DAC_VOL（地址0xc045）支持左右声道独立调节（高8位左声道/低8位右声道），步进精度为0.5分贝（dB）。 左右声道8位数值的低4位用于设置
6 dB步进的衰减量，高4位则控制0.5 dB步进的衰减量。

dB	DAC_VOL	dB	DAC_VOL	dB	DAC_VOL	dB	DAC_VOL
-0.0	0x0000	-6.5	0xb2b2	:	:	-60.0	0x0a0a
-0.5	0xb1b1	:	:	-36.0	0x0606	-60.5	0xbbbb
-1.0	0xa1a1	-12.0	0x0202	-36.5	0xb7b7	:	:
-1.5	0x9191	-12.5	0xb3b3	:	:	-66.0	0x0b0b
-2.0	0x8181	:	:	-42.0	0x0707	-66.5	0xbcbc
-2.5	0x7171	-18.0	0x0303	-42.5	0xb8b8	:	:
-3.0	0x6161	-18.5	0xb4b4	:	:	-72.0	0x0c0c
-3.5	0x5151	:	:	-48.0	0x0808	-72.5	0xbdbd
-4.0	0x4141	-24.0	0x0404	-48.5	0xb9b9	:	:
-4.5	0x3131	-24.5	0xb5b5	:	:	-78.0	0x0d0d
-5.0	0x2121	:	:	-54.0	0x0909	-78.5	0xbebe
-5.5	0x1111	-30.0	0x0505	-54.5	0xbaba	:	:
-6.0	0x0101	-30.5	0xb6b6	:	:	-84.0	0x0e0e

分贝	DAC_VOL	分贝	DAC_VOL	分贝	DAC_VOL	分贝	DAC_VOL
-0.0	0x0000	-6.5	0xb2b2	:	:	-60.0	0x0a0a
-0.5	0xb1b1	:	:	-36.0	0x0606	-60.5	0xbbbb
-1.0	0xa1a1	-12.0	0x0202	-36.5	0xb7b7	:	:
-1.5	0x9191	-12.5	0xb3b3	:	:	-66.0	0x0b0b
-2.0	0x8181	:	:	-42.0	0x0707	-66.5	0xbcbc
-2.5	0x7171	-18.0	0x0303	-42.5	0xb8b8	:	:
-3.0	0x6161	-18.5	0xb4b4	:	:	-72.0	0x0c0c
-3.5	0x5151	:	:	-48.0	0x0808	-72.5	0xbdbd
-4.0	0x4141	-24.0	0x0404	-48.5	0xb9b9	:	:
-4.5	0x3131	-24.5	0xb5b5	:	:	-78.0	0x0d0d
-5.0	0x2121	:	:	-54.0	0x0909	-78.5	0xbebe
-5.5	0x1111	-30.0	0x0505	-54.5	0xbaba	:	:
-6.0	0x0101	-30.5	0xb6b6	:	:	-84.0	0x0e0e

10.9 SPI Boot

If GPIO0 is set with a pull-up resistor to 1 at boot time, VS1053b tries to boot from external SPI memory.

SPI boot redefines the following pins:

Normal Mode	SPI Boot Mode
GPIO0	xCS
GPIO1	CLK
DREQ	MOSI
GPIO2	MISO

The memory has to be an SPI Bus Serial EEPROM with 16-bit or 24-bit addresses. The serial speed used by VS1053b is 245 kHz with the nominal 12.288 MHz clock. The first three bytes in the memory have to be 0x50, 0x26, 0x48.

10.10 Real-Time MIDI

If GPIO0 is low and GPIO1 is high during boot, real-time MIDI mode is activated. In this mode the PLL is configured to $4.0\times$, the UART is configured to the MIDI data rate 31250 bps, and real-time MIDI data is then read from UART and SDI. Both input methods should not be used simultaneously. If you use SDI, first send 0x00 and then send the MIDI data byte.

EarSpeaker setting can be configured with GPIO2 and GPIO3. The state of GPIO2 and GPIO3 are only read at startup.

Real-Time MIDI can also be started with a small patch code using SCI.

Note: The real-time MIDI parser in VS1053b does not know how to skip SysEx messages. An improved version can be loaded into IRAM if needed.

10.9 SPI引导

若启动时GPIO0通过上拉电阻设置为高电平（1），VS1053b将尝试从外部SPI存储器引导启动。

SPI引导模式将重新定义下列引脚功能：

正常模式	SPI引导模式
GPIO0	xCS
通用输入输出1	CLK
数据请求	MOSI
GPIO2	MISO

存储器必须是采用16位或24位地址的SPI总线串行EEPROM。 在标称12.288 MHz时钟频率下，VS1053b使用的串行速率为245千赫兹。存储器首三个字节必须为0x50、0x26、0x48。

10.10 实时MIDI

若启动时GPIO0为低电平且GPIO1为高电平，则激活实时MIDI模式。此模式下锁相环配置为 $4.0\times$倍频，UART配置为MIDI数据速率31250 bps，实时MIDI数据通过UART和SDI读取。两种输入方式不应同时使用。若使用SDI接口，请先发送0x00，再发送MIDI数据字节。

耳机/扬声器设置可通过GPIO2和GPIO3进行配置。GPIO2与GPIO3的状态仅在启动时读取。

实时MIDI功能也可通过SCI加载小型补丁代码启动。

注意：VS1053b中的实时MIDI解析器无法跳过系统专用信息(SysEx)。 如需改进版本，可将其加载至IRAM中。

10.11 Extra Parameters

The following structure is in X memory at address 0x1e02 (note the different location than in VS1033) and can be used to change some extra parameters or get useful information.

```
#define PARAMETRIC_VERSION 0x0003
struct parametric {
  /* configs are not cleared between files */
  u_int16 version;    /*1e02 - structure version */
  u_int16 config1;    /*1e03 ---- ---- ppss RRRR  PS mode, SBR mode, Reverb */
  u_int16 playSpeed;  /*1e04 0,1 = normal speed, 2 = twice, 3 = three times etc. */
  u_int16 byteRate;   /*1e05 average byterate */

  u_int16 endFillByte;    /*1e06 byte value to send after file sent */
  u_int16 reserved[16];   /*1e07..15 file byte offsets */
  u_int32 jumpPoints[8];  /*1e16..25 file byte offsets */
  u_int16 latestJump;  /*1e26 index to lastly updated jumpPoint */
  u_int32 positionMsec /*1e27-28 play position, if known (WMA, Ogg Vorbis) */
  s_int16 resync;      /*1e29 > 0 for automatic m4a, ADIF, WMA resyncs */
  union {
    struct {
      u_int32 curPacketSize;
      u_int32 packetSize;
    } wma;
    struct {
      u_int16 sceFoundMask; /*1e2a SCE's found since last clear */
      u_int16 cpeFoundMask; /*1e2b CPE's found since last clear */
      u_int16 lfeFoundMask; /*1e2c LFE's found since last clear */
      u_int16 playSelect;   /*1e2d 0 = first any, initialized at aac init */
      s_int16 dynCompress;  /*1e2e -8192=1.0, initialized at aac init */
      s_int16 dynBoost;     /*1e2f  8192=1.0, initialized at aac init */
      u_int16 sbrAndPsStatus; /*0x1e30 1=SBR, 2=upsample, 4=PS, 8=PS active */
    } aac;
    struct {
      u_int32 bytesLeft;
    } midi;
    struct {
      s_int16 gain; /* 0x1e2a proposed gain offset in 0.5dB steps, default = -12 */
    } vorbis;
  } i;
};
```

Notice that reading two-word variables through the SCI_WRAMADDR and SCI_WRAM interface is not protected in any way. The variable can be updated between the read of the low and high parts. The problem arises when both the low and high parts change values. To determine if the value is correct, you should read the value twice and compare the results.

The following example shows what happens when `bytesLeft` is decreased from 0x10000 to 0xffff and the update happens between low and high part reads or after high part read.

Read Invalid		Read Valid		No Update	
Address	**Value**	**Address**	**Value**	**Address**	**Value**
0x1e2a	0x0000 change after this	0x1e2a	0x0000	0x1e2a	0x0000
0x1e2b	0x0000	0x1e2b	0x0001 change after this	0x1e2b	0x0001
0x1e2a	0xffff	0x1e2a	0xffff	0x1e2a	0x0000
0x1e2b	0x0000	0x1e2b	0x0000	0x1e2b	0x0001

10.11 扩展参数

以下结构体位于X存储器地址0x1e02处（注意：与VS1033的位置不同），可用于修改额外参数或获取有用信息。

```
#define PARAMETRIC_VERSION 0x0003
struct parametric {
  /* 配置文件在文件之间不会被清除 */ u_int16 versi
  on; /*1e02 - 结构体版本号 */
  u_int16 config1;          /*1e03 ---- ----  ---- ---- -ppss RRRR PS模式、SBR模式、混响效果 */
  u_int16 playSpeed;        /*1e04 0,1=正常速度, 2=两倍速, 3=三倍速等 */
  u_int16 byteRate;         /*1e05 平均字节速率 */

  u_int16 endFillByte;          /*1e06 文件发送后需传输的字节值 */
  u_int16 reserved[16];         /*1e07..15 文件字节偏移量 */
  u_int32 jumpPoints[8];        /*1e16..25 文件字节偏移量 */
  u_int16 latestJump;       /*1e26 最后更新的跳转点索引 */
  u_int32 positionMsec /*1e27-28 已知播放位置（WMA, Ogg Vorbis 格式）*/
  s_int16 resync;               /*1e29 >0 表示自动进行m4a/ADIF/WMA重新同步 */
  union {
    struct {
      u_int32 curPacketSize;
      u_int32 packetSize;
    } wma;
    struct {
      u_int16 sceFoundMask; /*1e2a 自上次清除后检测到的SCE数量 */
      u_int16 cpeFoundMask; /*1e2b 自上次清除后检测到的CPE数量 */
      u_int16 lfeFoundMask; /*1e2c 自上次清除后检测到的LFE数量 */
      u_int16 playSelect;         /*1e2d 0=首选任意通道, 初始化于AAC初始化阶段 */
      s_int16 dynCompress;      /*1e2e -8192=1.0, 初始化于AAC初始化阶段 */
      s_int16 dynBoost;           /*1e2f 8192=1.0, 初始化于AAC初始化阶段 */
      u_int16 sbrAndPsStatus; /*0x1e30 1=SBR, 2=上采样, 4=PS, 8=PS激活 */ } aac;

    struct {
      u_int32 bytesLeft;
    } midi;
    struct {
      s_int16 gain; /* 0x1e2a 建议增益偏移量（0.5dB步进），默认值 = -12 */ } vorbis; } i; };
```

请注意，通过SCI_WRAMADDR和SCI_WRAM接口读取双字变量时未采取任何保护机制。该变量可能在读取低位部分与高位部分之间被更新。当低位部分和高位部分同时发生值变更时，问题即会产生。为确保数值正确性，应对该值执行两次读取并比较结果。

以下示例演示当 bytesLeft从0x10000递减至0xffff，且更新操作发生在低位/高位读取之间或高位读取之后的情形。

读取无效		读取有效		无更新	
地址	**取值**	**地址**	**取值**	**地址**	**取值**
0x1e2a 0x1e2b	0x0000 此后变更 0x0000	0x1e2a 0x1e2b	0x0000 0x0001 此后变更	0x1e2a 0x1e2b	0x0000 0x0001
0x1e2a 0x1e2b	0xffff 0x0000	0x1e2a 0x1e2b	0xffff 0x0000	0x1e2a 0x1e2b	0x0000 0x0001

You can see that in the invalid read the low part wraps from 0x0000 to 0xffff while the high part stays the same. In this case the second read gives a valid answer, otherwise always use the value of the first read. The second read is needed when it is possible that the low part wraps around, changing the high part, i.e. when the low part is small. `bytesLeft` is only decreased by one at a time, so a reread is needed only if the low part is 0.

10.11.1 Common Parameters

These parameters are common for all codecs. Other fields are only valid when the corresponding codec is active. The currently active codec can be determined from SCI_HDAT1.

Parameter	Address	Usage
version	0x1e02	Structure version – 0x0003
config1	0x1e03	Miscellaneous configuration
playSpeed	0x1e04	0,1 = normal speed, 2 = twice, 3 = three times etc.
byteRate	0x1e05	average byterate
endFillByte	0x1e06	byte to send after file
jumpPoints[8]	0x1e16-25	Packet offsets for WMA and AAC
latestJump	0x1e26	Index to latest jumpPoint
positionMsec	0x1e27-28	File position in milliseconds, if available
resync	0x1e29	Automatic resync selector

The fuse-programmed ID is read at startup and copied into the `chipID` field. If not available, the value will be all zeros. The `version` field can be used to determine the layout of the rest of the structure. The version number is changed when the structure is changed. For VS1053b the structure version is 3.

`config1` controls MIDI Reverb and AAC's SBR and PS settings.

`playSpeed` makes it possible to fast forward songs. Decoding of the bitstream is performed, but only each `playSpeed` frames are played. For example by writing 4 to `playSpeed` will play the song four times as fast as normal, if you are able to feed the data with that speed. Write 0 or 1 to return to normal speed. SCI_DECODE_TIME will also count faster. All current codecs support the `playSpeed` configuration.

`byteRate` contains the average bitrate in bytes per second for every code. The value is updated once per second and it can be used to calculate an estimate of the remaining playtime. This value is also available in SCI_HDAT0 for all codecs except MP3, MP2, and MP1.

`endFillByte` indicates what byte value to send after file is sent before SM_CANCEL.

`jumpPoints` contain 32-bit file offsets. Each valid (non-zero) entry indicates a start of a packet for WMA or start of a raw data block for AAC (ADIF, .mp4 / .m4a). `latestJump` contains the index of the entry that was updated last. If you only read entry pointed to by `latestJump` you do *not* need to read the entry twice to ensure validity. Jump point information can be used to

可见在无效读取时，低位部分会从0x0000回绕至0xffff，而高位部分保持不变。此种情况下第二次读取将获得有效值，否则应始终采用首次读取的数值。当低位部分可能发生回绕并改变高位值时（即低位数值较小时），需进行二次读取。 bytesLeft每次仅递减1，因此仅当低位为0时才需要重新读取。

10.11.1 通用参数

这些参数对所有编解码器通用。其余字段仅在对应编解码器激活时有效。当前激活的编解码器可通过SCI_HDAT1寄存器确定。

参数	地址	用途
版本	0x1e02	结构版本 – 0x0003
配置1	0x1e03	杂项配置
播放速度	0x1e04	0,1=正常速度, 2=两倍速, 3=三倍速（以此类推）
字节率	0x1e05	平均字节率
结束填充字节	0x1e06	文件结束后发送的字节
跳转点数组[8]	0x1e16-25	WMA和AAC的数据包偏移量
最新跳转点	0x1e26	最新跳转点的索引
位置毫秒数	0x1e27-28	文件位置（毫秒单位，如可用）
重新同步	0x1e29	自动重新同步选择器

熔丝编程的ID在启动时被读取并复制到芯片ID(chipID)字段中。若不可用，该值将全为零。版本(version)字段可用于确定结构其余部分的布局。结构变更时版本号将随之更新。VS1053b的结构版本号为3。

配置寄存器1(config1)控制MIDI混响及AAC的SBR（频谱带复制）与PS（参数立体声）设置。

播放速度(playSpeed)支持歌曲快进功能。系统仍执行比特流解码，但仅播放每播放速度(playSpeed)帧的数据。例如：向播放速度(playSpeed)写入4，将以四倍正常速度播放歌曲（前提是能以该速率提供数据）。 写入0或1可恢复正常速度。SCI_DECODE_TIME寄存器的计数速度也将同步加快。所有当前编解码器均支持播放速度(playSpeed)配置。

字节率包含每种编码的平均比特率（单位：字节/秒）。 该数值每秒更新一次，可用于估算剩余播放时间。除MP3、MP2和MP1外，所有编码格式均可通过SCI_HDAT0寄存器获取此值。

结束填充字节指示文件发送完成后、SM_CANCEL执行前需发送的字节值。

跳转点存储32位文件偏移量。每个有效（非零）条目标识WMA数据包起始点或AAC（ADIF, .mp4/.m4a）原始数据块起始点。最新跳转存储最近更新的条目索引。若仅读取最新跳转指向的条目，则无需重复读取即可确保数据有效性。跳转点信息可用于

implement perfect fast forward and rewind for WMA and AAC (ADIF, .mp4 / .m4a).

positionMsec is a field that gives the current play position in a file in milliseconds, regardless of rewind and fast forward operations. The value is only available in codecs that can determine the play position from the stream itself. Currently WMA and Ogg Vorbis provide this information. If the position is unknown, this field contains -1.

resync field is used to force a resynchronization to the stream for WMA and AAC (ADIF, .mp4 / .m4a) instead of ending the decode at first error. This field can be used to implement almost perfect fast forward and rewind for WMA and AAC (ADIF, .mp4 / .m4a). The user should set this field before performing data seeks if they are not in packet or data block boundaries. The field value tells how many tries are allowed before giving up. The value 32767 gives infinite tries.

The resync field is set to 32767 after a reset to make resynchronization the default action, but it can be cleared after reset to restore the old action. When resync is set, every file decode should always end as described in Chapter 10.5.1.

Seek fields no longer exist. When resync is required, WMA and AAC codecs now enter broadcast/stream mode where file size information is ignored. Also, the file size and sample size information of WAV files are ignored when resync is non-zero. The user must use SM_CANCEL or software reset to end decoding.

Note: WAV, WMA, ADIF, and .mp4 / .m4a files begin with a metadata or header section, which must be fully processed before any fast forward or rewind operation. SS_DO_NOT_JUMP (in SCI_STATUS) is clear when the header information has been processed and jumps are allowed.

10.11.2 WMA

Parameter	Address	Usage
curPacketSize	0x1e2a/2b	The size of the packet being processed
packetSize	0x1e2c/2d	The packet size in ASF header

The ASF header packet size is available in packetSize. With this information and a packet start offset from jumpPoints you can parse the packet headers and skip packets in ASF files.

WMA decoder can also increase the internal clock automatically when it detects that a file can not be decoded correctly with the current clock. The maximum allowed clock is configured with the SCI_CLOCKF register.

实现WMA及AAC（ADIF, .mp4/.m4a）格式的完美快进与快退功能。

positionMsec字段表示当前文件播放位置（单位：毫秒），该数值不受快退/快进操作影响。仅当编解码器可从数据流自身确定播放位置时，此数值才有效。当前仅WMA和Ogg Vorbis格式提供此信息。
若位置未知，此字段值为-1。

resync字段用于强制WMA和AAC（ADIF, .mp4/.m4a）格式的数据流重新同步，而非在首次出现错误时终止解码。该字段可实现WMA和AAC（ADIF, .mp4/.m4a）近乎完美的快进/快退功能。若数据查找不在数据包或数据块边界时，用户应在执行查找前设置此字段。字段值表示放弃前允许的重试次数。值32767表示无限次重试。

复位后，再同步字段默认设为32767以启用再同步操作，但复位后可清除该值以恢复旧有行为。启用再同步时，每次文件解码必须严格遵循第10.5.1章所述流程结束。

定位字段已取消。需要再同步时，WMA与AAC编解码器将进入广播/流模式，此模式下忽略文件大小信息。当再同步值非零时，WAV音频格式文件的文件大小与采样大小信息同样会被忽略。用户必须通过SM_CANCEL指令或软件复位终止解码过程。

注意：WAV、WMA、ADIF及.mp4/.m4a文件起始部分包含元数据或头部信息区段，执行快进或倒退操作前必须完整处理该区段。当头部信息处理完毕且允许跳转时，SCI_STATUS寄存器中的SS_DO_NOT_JUMP标志位将清零。

10.11.2 WMA

参数	地址	用途
curPacketSize	0x1e2a/2b	当前正在处理的数据包大小
packetSize	0x1e2c/2d	ASF文件头中的数据包大小

ASF文件头数据包大小可在 packetSize中获取。结合此信息及 jumpPoints中的数据包起始偏移量，可解析ASF文件的数据包头并跳过数据包。

当检测到当前时钟无法正确解码文件时，WMA解码器也可自动提升内部时钟频率。允许的最大时钟频率通过SCI_CLOCKF寄存器配置。

10.11.3 AAC

Parameter	Address	Usage
config1	0x1e03(7:4)	SBR and PS select
sceFoundMask	0x1e2a	Single channel elements found
cpeFoundMask	0x1e2b	Channel pair elements found
lfeFoundMask	0x1e2c	Low frequency elements found
playSelect	0x1e2d	Play element selection
dynCompress	0x1e2e	Compress coefficient for DRC, -8192=1.0
dynBoost	0x1e2f	Boost coefficient for DRC, 8192=1.0
sbrAndPsStatus	0x1e30	SBR and PS available flags

`playSelect` determines which element to decode if a stream has multiple elements. The value is set to 0 each time AAC decoding starts, which causes the first element that appears in the stream to be selected for decoding. Other values are: 0x01 - select first single channel element (SCE), 0x02 - select first channel pair element (CPE), 0x03 - select first low frequency element (LFE), $S * 16 + 5$ - select SCE number S, $P * 16 + 6$ - select CPE number P, $L * 16 + 7$ - select LFE number L. When automatic selection has been performed, `playSelect` reflects the selected element.

`sceFoundMask`, `cpeFoundMask`, and `lfeFoundMask` indicate which elements have been found in an AAC stream since the variables have last been cleared. The values can be used to present an element selection menu with only the available elements.

`dynCompress` and `dynBoost` change the behavior of the dynamic range control (DRC) that is present in some AAC streams. These are also initialized when AAC decoding starts.

`sbrAndPsStatus` indicates spectral band replication (SBR) and parametric stereo (PS) status.

Bit	Usage
0	SBR present
1	upsampling active
2	PS present
3	PS active

Bits 7 to 4 in `config1` can be used to control the SBR and PS decoding. Bits 5 and 4 select SBR mode and bits 7 and 6 select PS mode. These configuration bits are useful if your AAC license does not cover SBR and/or PS.

config1(5:4)	Usage
'00'	normal mode, upsample $<$24 kHz AAC files
'01'	do not automatically upsample $<$24 kHz AAC files, but enable upsampling if SBR is encountered
'10'	never upsample
'11'	disable SBR (also disables PS)

10.11.3 AAC

参数	地址	用途
配置1	0x1e03(7:4)	SBR与PS选择
sceFoundMask	0x1e2a	检测到的单声道元素
cpeFoundMask	0x1e2b	检测到的声道对元素
lfeFoundMask	0x1e2c	检测到低频元素
播放选择	0x1e2d	播放元素选择
动态压缩	0x1e2e	DRC压缩系数，-8192=1.0
动态增强	0x1e2f	DRC增强系数，8192=1.0
SBR与PS状态	0x1e30	SBR及PS可用标志位

playSelect决定当流包含多元素时需解码的元素每次启动AAC解码时该值重置为0，此时将选择流中首个出现的元素进行解码其他取值：0x01 - 选择首个单声道元素(SCE)，0x02 - 选择首个声道对元素(CPE)，0x03 - 选择首个低频元素

(LFE), $S\times16+5$ - 选择第S个SCE, $P\times16+6$ - 选择第P个CPE, $L\times16+7$ - 选择第L个LFE。自动选择执行后，play$S$elect将反映所选元素

变量cpeFoundMask、lfeFoundMask与sceFoundMask用于表明自上次清除以来，AAC流中存在哪些编码元素。这些值可用于生成仅包含可用元素的编码器选择菜单。

dynCompress与 dynBoost参数可调整部分AAC流中动态范围控制(DRC)的功能特性。这些参数在AAC解码启动时也会被初始化。

sbrAndPsStatus 指示频带复制(SBR)与参数化立体声(PS)的状态信息。

位域	用途
0	存在SBR
1	上采样激活中
2	存在PS
3	PS激活中

config1寄存器的位7至4可用于控制SBR与PS解码过程。位5和4选择SBR模式，位7和6选择PS模式。若您的AAC授权未涵盖SBR和/或PS功能，这些配置位将发挥重要作用。

config1(5:4)位域	用途
’00’	正常模式，上采样 <24 kHz AAC 文件
’01’	不自动上采样 <24 kHz AAC 文件，但若检测到SBR则启用上采样
’10’	永不进行上采样
’11’	禁用SBR（同时禁用PS）

config1(7:6)	Usage
'00'	normal mode, process PS if it is available
'01'	process PS if it is available, but in downsampled mode
'10'	reserved
'11'	disable PS processing

AAC decoder can also increase the internal clock automatically when it detects that a file can not be decoded correctly with the current clock. The maximum allowed clock is configured with the SCI_CLOCKF register.

If even the highest allowed clock is too slow to decode an AAC file with SBR and PS components, the advanced decoding features are automatically dropped one by one until the file can be played. First the parametric stereo processing is dropped (the playback becomes mono). If that is not enough, the spectral band replication is turned into downsampled mode (reduced bandwidth). As the last resort the spectral band replication is fully disabled. Dropped features are restored at each song change.

10.11.4 Midi

Parameter	Address	Usage
config1	0x1e03	Miscellaneous configuration
	bits [3:0]	Reverb: 0 = auto (ON if clock $>= 3.0\times$) 1 = off, 2 - 15 = room size
bytesLeft	0x1e2a/2b	The number of bytes left in this track

The lowest 4 bits of `config1` controls the reverb effect.

10.11.5 Ogg Vorbis

Parameter	Address	Usage
gain	0x1e2a	Preferred replay-gain offset

Ogg Vorbis decoding supports Replay Gain technology. The Replay Gain technology is used to automatically give all songs a matching volume so that the user does not need to adjust the volume setting between songs. If the Ogg Vorbis decoder finds a Replay Gain tag in the song header, the tag is parsed and the decoded gain setting can be found from the `gain` parameter. For a song without any Replay Gain tag, a default of -6 dB (`gain` value -12) is used. For more details about Replay Gain, see *http://en.wikipedia.org/wiki/Replay_Gain* and *http://www.replaygain.org/*.

The player software can use the gain value to adjust the volume level. Negative values mean that the volume should be decreased, positive values mean that the volume should be increased.

配置寄存器1(7:6)	用途
’00’	正常模式，若可用则处理PS
’01’	若可用则处理PS，但采用下采样模式
’10’	保留
’11’	禁用PS处理

当检测到文件无法在当前时钟下正确解码时，AAC解码器可自动提升内部时钟频率。允许的最大时钟频率通过SCI_CLOCKF寄存器配置。

若即使最高允许时钟仍不足以解码含SBR和PS组件的AAC文件，系统将逐级放弃高级解码功能直至文件可正常播放。首先放弃参数立体声处理（播放转为单声道）。

若仍不足，则将频带复制转为下采样模式（降低带宽）。 作为最后手段，频谱带复制功能将被完全禁用。每次切换歌曲时，被舍弃的功能会恢复。

10.11.4 MIDI格式

参数	地址	用途
配置1	0x1e03	杂项配置
	位[3:0]	混响：0=自动（当clock >= 3.0×时启用） 1=关闭，2-15=空间尺寸
剩余字节数	0x1e2a/2b	当前音轨剩余字节数量

config1寄存器的最低4位控制混响效果。

10.11.5 Ogg Vorbis

参数	地址	用途
增益值	0x1e2a	首选回放增益偏移量

Ogg Vorbis格式解码支持回放增益技术。回放增益技术可自动匹配所有歌曲音量，用户无需在歌曲间手动调整音量设置。若Ogg Vorbis解码器在歌曲头部发现回放增益标签，将解析该标签并可通过增益参数获取解码后的增益设置。对于不含回放增益标签的歌曲，默认采用-6 dB（对应增益值-12）。 有关Replay Gain的更多详情，请参见http://en.wikipedia.org/wiki/Replay_Gain及 http://www.replaygain.org/。

播放器软件可利用增益值调节音量级别。负值表示应降低音量，正值表示应提高音量。

For example `gain` = -11 means that volume should be decreased by 5.5 dB ($-11/2 = -5.5$), and left and right attenuation should be increased by 11. When `gain` = 2 volume should be increased by 1 dB ($2/2 = 1.0$), and left and right attenuation should be decreased by 2. Because volume setting can not go above +0 dB, the value should be saturated.

Gain	Volume	SCI_VOL (Volume-Gain)
-11 (-5.5 dB)	0 (+0.0 dB)	0x0b0b (-5.5 dB)
-11 (-5.5 dB)	3 (-1.5 dB)	0x0e0e (-7.0 dB)
+2 (+1.0 dB)	0 (+0.0 dB)	0x0000 (+0.0 dB)
+2 (+1.0 dB)	1 (-0.5 dB)	0x0000 (+0.0 dB)
+2 (+1.0 dB)	4 (-2.0 dB)	0x0202 (-1.0 dB)

例如增益= -11 表示音量应降低5.5 dB $(-11/2 = -5.5)$,
且左右声道衰减量应增加11个单位。当增益= 2 时，音量应提高1 dB $(2/2 = 1.0)$，同时左右声道衰减量应减少2个单位。由于音量设置不可超过+0 dB，该数值应进行饱和处理。

增益	音量	SCI_VOL (音量-增益)
-11 (-5.5 dB)	0 (+0.0 dB)	0x0b0b (-5.5 dB)
-11 (-5.5 dB)	3 (-1.5 dB)	0x0e0e (-7.0 dB)
+2 (+1.0 dB)	0 (+0.0 dB)	0x0000 (+0.0 dB)
+2 (+1.0 dB)	1 (-0.5 dB)	0x0000 (+0.0 dB)
+2 (+1.0 dB)	4 (-2.0 dB)	0x0202 (-1.0 dB)

10.12 SDI Tests

There are several test modes in VS1053b, which allow the user to perform memory tests, SCI bus tests, and several different sine wave tests.

All tests are started in a similar way: VS1053b is hardware reset, SM_TESTS is set, and then a test command is sent to the SDI bus. Each test is started by sending a 4-byte special command sequence, followed by 4 zeros. The sequences are described below.

10.12.1 Sine Test

Sine test is initialized with the 8-byte sequence 0x53 0xEF 0x6E n 0 0 0 0, where n defines the sine test to use. n is defined as follows:

n **bits**		
Name	**Bits**	**Description**
F_sIdx	7:5	Samplerate index
S	4:0	Sine skip speed

F_sIdx	F_s	F_sIdx	F_s
0	44100 Hz	4	24000 Hz
1	48000 Hz	5	16000 Hz
2	32000 Hz	6	11025 Hz
3	22050 Hz	7	12000 Hz

The frequency of the sine to be output can now be calculated from $F = F_s \times \frac{S}{128}$.

Example: Sine test is activated with value 126, which is 0b01111110. Breaking n to its components, $F_sIdx = 0b011 = 3$ and thus $F_s = 22050Hz$. $S = 0b11110 = 30$, and thus the final sine frequency $F = 22050Hz \times \frac{30}{128} \approx 5168Hz$.

To exit the sine test, send the sequence 0x45 0x78 0x69 0x74 0 0 0 0.

Note: Sine test signals go through the digital volume control, so it is possible to test channels separately.

10.12.2 Pin Test

Pin test is activated with the 8-byte sequence 0x50 0xED 0x6E 0x54 0 0 0 0. This test is meant for chip production testing only.

10.12.3 SCI Test

Sci test is initialized with the 8-byte sequence 0x53 0x70 0xEE n 0 0 0 0, where n is the register number to test. The content of the given register is read and copied to SCI_HDAT0. If

10.12 SDI测试项

VS1053b 包含多种测试模式，用户可执行存储器测试、SCI总线测试及多种不同的正弦波测试。

所有测试启动方式相似：先对VS1053b进行硬件复位，设置SM_TESTS标志位，然后将测试命令发送至SDI总线。每个测试通过发送4字节特殊命令序列启动，后跟4个零值字节。具体命令序列说明如下。

10.12.1 正弦测试

正弦测试通过8字节序列 0x53 0xEF 0x6E n0 0 0 0 初始化，其中 n定义所用正弦测试类型。 n 取值定义如下：

n 位		
名称	**位**	**描述**
F_s索引	7:5	采样率索引
S	4:0	正弦波跳过速度

F_s索引	F_s	F_s索引	F_s
0	44100 赫兹	4	24000 赫兹
1	48000 赫兹	5	16000 赫兹
2	32000 赫兹	6	11025 赫兹
3	22050 赫兹	7	12000 赫兹

待输出正弦波的频率可通过公式计算： $F = F_s \times \frac{S}{128}$。

示例：正弦测试以数值126激活，即$0b01111110$。 将 n分解为分量， F_s索引$= 0b011 = 3$，因此Fs$=$ 22050赫兹。S$=$ $0b11110 = 30$，因此最终正弦频率F$=$ 22050赫兹$\times \frac{30}{128} \approx 5168$赫兹。

退出正弦测试需发送序列：0x45 0x78 0x69 0x74 0 0 0 0。

注：正弦测试信号经由数字音量控制器处理，故可单独测试各声道。

10.12.2 引脚测试

引脚测试通过8字节序列 0x50 0xED 0x6E 0x54 0 0 0 0 激活。此测试仅用于芯片生产检测。

10.12.3 SCI测试

Sci测试通过8字节序列 0x53 0x70 0xEE n0 0 0 0 初始化，其中 n为待测试寄存器编号。指定寄存器的内容将被读取并复制至SCI_HDAT0。若

the register to be tested is HDAT0, the result is copied to SCI_HDAT1.

Example: if n is 0, contents of SCI register 0 (SCI_MODE) is copied to SCI_HDAT0.

10.12.4 Memory Test

Memory test mode is initialized with the 8-byte sequence 0x4D 0xEA 0x6D 0x54 0 0 0 0. After this sequence, wait for 1100000 clock cycles. The result can be read from the SCI register SCI_HDAT0, and 'one' bits are interpreted as follows:

Bit(s)	Mask	Meaning
15	0x8000	Test finished
14:10		Unused
9	0x0200	Mux test succeeded
8	0x0100	Good MAC RAM
7	0x0080	Good I RAM
6	0x0040	Good Y RAM
5	0x0020	Good X RAM
4	0x0010	Good I ROM 1
3	0x0008	Good I ROM 2
2	0x0004	Good Y ROM
1	0x0002	Good X ROM 1
0	0x0001	Good X ROM 2
	0x83ff	All ok

Memory tests overwrite the current contents of the RAM memories.

10.12.5 New Sine and Sweep Tests

A more frequency-accurate sine test can be started and controlled from SCI. SCI_AICTRL0 and SCI_AICTRL1 set the sine frequencies for left and right channel, respectively. These registers, volume (SCI_VOL), and samplerate (SCI_AUDATA) can be set before or during the test. Write 0x4020 to SCI_AIADDR to start the test.

SCI_AICTRLn can be calculated from the desired frequency and DAC samplerate by:

$$SCI\_AICTRLn = F_{sin} \times 65536/F_s$$

The maximum value for SCI_AICTRLn is 0x8000U. For the best S/N ratio for the generated sine, three LSb's of the SCI_AICTRLn should be zero. The resulting frequencies F_{sin} can be calculated from the DAC samplerate F_s and SCI_AICTRL0 / SCI_AICTRL1 using the following equation.

被测寄存器为HDAT0，测试结果将复制至SCI_HDAT1寄存器。

示例：若 n为0，则SCI寄存器0（SCI_MODE）的内容将被复制到SCI_HDAT0。

10.12.4 存储器测试

内存测试模式以8字节序列0x4D 0xEA 0x6D 0x54 0 0 0 0进行初始化。此序列结束后，需等待1 100000个时钟周期。结果可从SCI寄存器SCI_HDAT0读取，其中置位位的含义如下：

位域	掩码	含义
15	0x8000	测试完成
14:10		未使用
9	0x0200	多路复用器测试通过
8	0x0100	MAC RAM正常
7	0x0080	指令RAM正常
6	0x0040	数据RAM正常
5	0x0020	X RAM良好
4	0x0010	I ROM 1良好
3	0x0008	I ROM 2良好
2	0x0004	Y ROM良好
1	0x0002	X ROM 1良好
0	0x0001	X ROM 2良好
	0x83ff	全部正常

内存测试会覆盖RAM存储器当前内容。

10.12.5 新型正弦与扫频测试

可通过SCI启动并控制频率更精确的正弦测试。SCI_AICTRL0和SCI_AICTRL1寄存器分别设置左右声道的正弦波频率。这些寄存器、音量(SCI_VOL)和采样率(SCI_AUDATA)可在测试前或测试过程中设置。向SCI_AIADDR寄存器写入0x4020以启动测试。

SCI_AICTRLn可根据目标频率和DAC采样率通过下式计算：

$$SCI\_AICT\ RLn = F_{sin} \times 65536 / F_s$$

SCI_AICTRLn 寄存器的最大值为 0x8000U。 为获得生成正弦波的最佳信噪比，SCI_AICTRLn 的三个最低有效位应置零。生成的正弦波频率 $F_s$in 可通过 DAC 采样率 F_s 及 SCI_AICTRL0 / SCI_AICTRL1 寄存器值，依据以下公式计算得出：

$$F_{sin} = SCI\_AICTRLn \times F_s/65536$$

Sine sweep test can be started by writing 0x4022 to SCI_AIADDR.

Both these tests use the normal audio path, thus also SCI_BASS, differential output mode, and EarSpeaker settings have an effect.

$$F\_sin = SCI\_AICTRL_n \times F\_s \,/\, 65536$$

可通过向 SCI_AIADDR 寄存器写入 0x4022 启动正弦扫频测试。

上述测试均使用常规音频通路，因此 SCI_BASS 寄存器、差分输出模式及耳机/扬声器设置均会影响测试结果。

11 VS1053b Registers

11.1 Who Needs to Read This Chapter

User software is required when a user wishes to add some own functionality like DSP effects to VS1053b.

However, most users of VS1053b don't need to worry about writing their own code, or about this chapter, including those who only download software plug-ins from VLSI Solution's Web site.

Note: Also see VS1063 Hardware Guide for more information, because the hardware is compatible with VS1053.

11.2 The Processor Core

VS_DSP is a 16/32-bit DSP processor core that also had extensive all-purpose processor features. VLSI Solution's free VSKIT Software Package contains all the tools and documentation needed to write, simulate and debug Assembly Language or Extended ANSI C programs for the VS_DSP processor core. VLSI Solution also offers a full Integrated Development Environment VSIDE for full debug capabilities.

11 VS1053b寄存器

11.1 本章阅读对象

当用户需要为 VS1053b 添加自定义功能（如 DSP 特效）时，需开发相应软件。

然而，大多数 VS1053b 用户（包括仅从 VLSI Solution 官网下载软件插件的用户）无需自行编写代码，也无需关注本章内容。

注：更多信息请另见VS1063硬件指南，因为其硬件与VS1053兼容。

11.2 处理器核心

VS_DSP音频处理器是一个16/32位DSP处理器核心，同时具备全面的通用处理器特性。VLSI Solution的免费VSKIT软件包包含所有必要的工具和文档，可用于为VS_DSP处理器核心编写、模拟及调试汇编语言或扩展ANSI C程序。VLSI Solution同时提供完整的集成开发环境VSIDE，具备全面调试功能。

11.3 VS1053b Hardware DAC Audio Paths

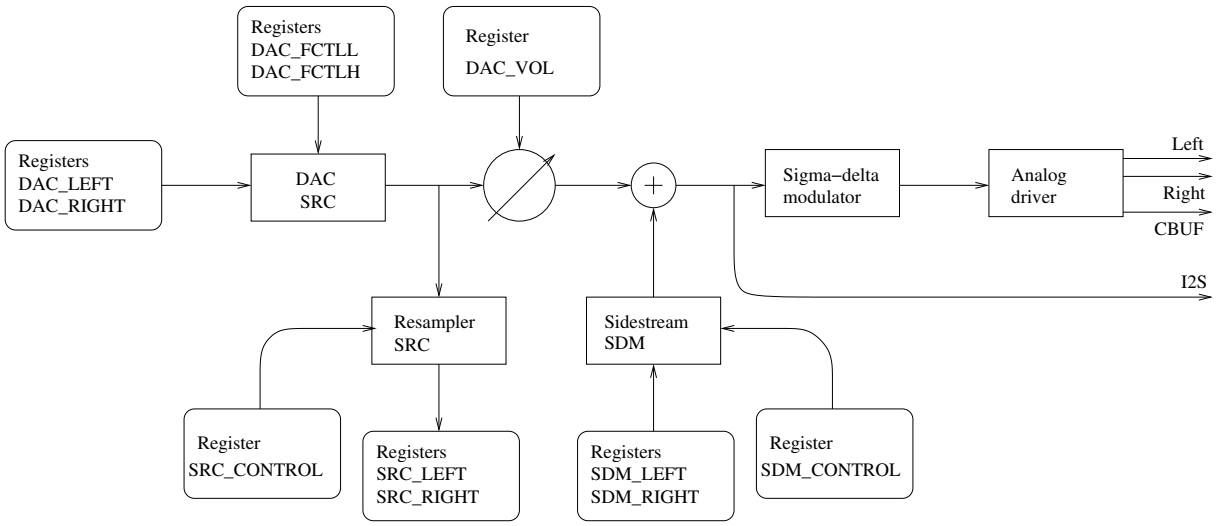


Figure 18: VS1053b ADC and DAC data paths with some data registers

Figure 18 presents the VS1053b Hardware DAC audio paths.

The main audio path starts from the DAC register (Chapter 11.8) to the high-fidelity, fully digital DAC SRC (Digital-to-Analog Converter SampleRate Converter), which low-pass filters and interpolates the data to the high samplerate of XTALI/2 (nominally 6.144 MHz). This 18-bit data is then fed to the volume control. It then passes through the sigma-delta modulator to the analog driver and analog Left and Right signals.

The user may resample and record the data with the Resampler SampleRate Converter (Chapter 11.16). Because there is no automatic low-pass filtering, it is the user's responsibility to avoid aliasing distortion.

The user may add a PCM sidestream with the Sidestream Sigma-Delta Modulator input (Chapter 11.17). As is the case with the Resampler SampleRate Converter, hardware doesn't offer low-pass filtering, so sufficient aliasing image rejection is the responsibility of the user.

11.3 VS1053b硬件DAC音频路径

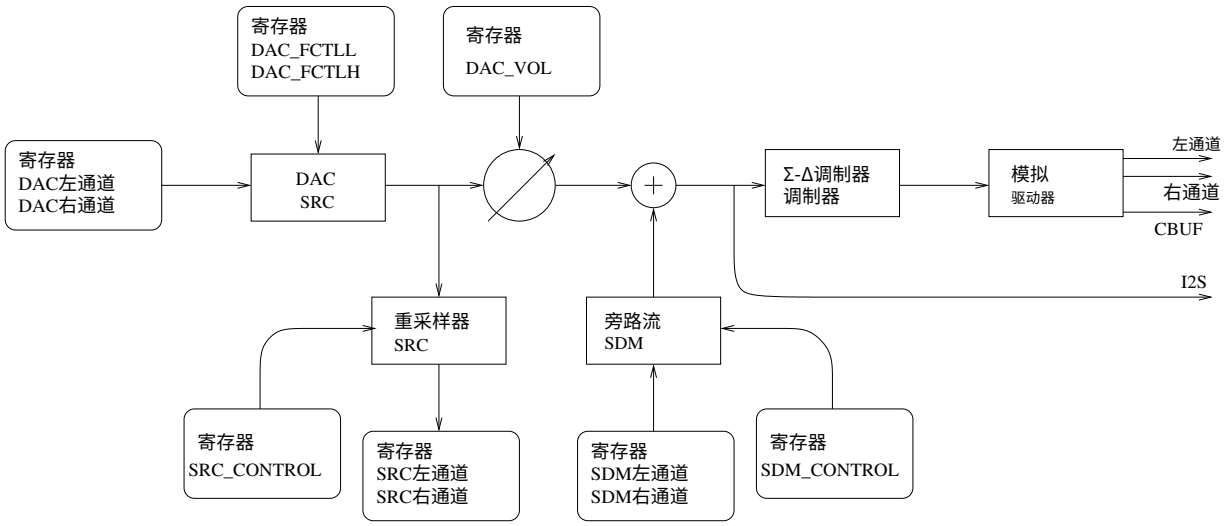


图18：包含部分数据寄存器的VS1053b ADC与DAC数据路径

图18展示了VS1053b硬件DAC音频路径

主音频路径起始于DAC寄存器（第11.8章），经高保真全数字DAC SRC（数模转换器采样率转换器）进行低通滤波和数据插值处理，提升至XTALI/2的高采样率（标称值6.144 MHz）。该18位数据随后送入音量控制器。数据接着通过Σ-Δ调制器传输至模拟驱动器，最终生成模拟左右声道信号。

用户可通过重采样器（第11.16章）对数据进行重采样和录制。由于未配置自动低通滤波，用户需自行避免混叠失真。

用户可通过旁流Σ-Δ调制器输入添加PCM旁路流（参见第11.17章）。与重采样器采样率转换器情况类似，硬件未提供低通滤波功能，因此充分的混叠镜像抑制需由用户自行实现。

11.4 VS1053b Hardware ADC Audio Paths

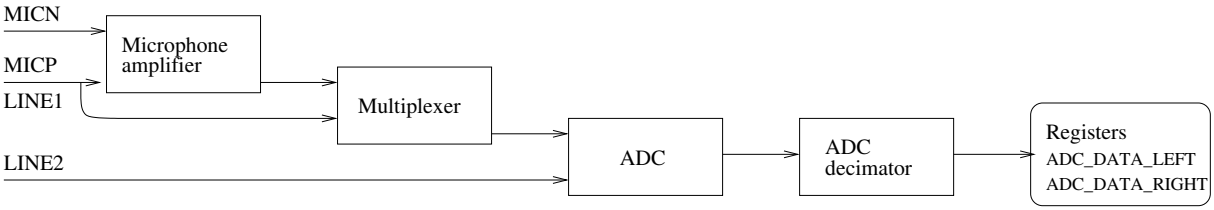


Figure 19: VS1053b ADC and DAC data paths with some data registers

Figure 18 presents the VS1053b Hardware ADC audio paths.

Analog audio may be fed upto two channels: one as a differential signal to MICN/MICP or as a one-sided signal to Line1, and the other as a one-sided signal to Line2.

If microphone input for the left channel has been selected, audio is fed through a microphone amplifier and that signal is selected by a multiplexer.

Audio is then downsampled to one of four allowed samplerates: XTALI/64, XTALI/128, XTALI/256 or XTALI/512. With the nominal 12.288 MHz crystal, these correspond to 192, 96, 48 or 24 kHz samplerates, respectively (Chapter 11.17).

If the “3 MHz” option bit SS_AD_CLOCK in register SCI_STATUS has been set to 1, then samplerates are divided by two, so the nominal samplerates become 96, 48, 24 and 12 kHz.

11.4 VS1053b硬件ADC音频通路

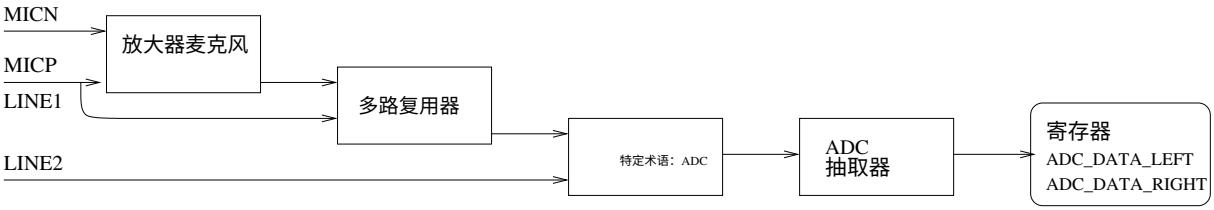


图19：VS1053b ADC与DAC数据通路及相关数据寄存器

图18展示了VS1053b硬件ADC音频通路结构。

模拟音频可输入双通道：差分信号至MICN/MICP或单端信号至Line1构成第一通道，单端信号至Line2构成第二通道。

若选择麦克风作为左声道输入，音频信号将经由麦克风放大器处理，并由多路复用器完成信号选择。

音频随后被下采样至四种允许的采样率之一：XTALI/64、XTALI/128、XTALI/256 或 XTALI/512。使用标称值12.288 MHz晶振时，这些分频分别对应192、96、48或24千赫兹采样率（第11.17章）。

若寄存器SCI_STATUS中的'3 MHz'选项位SS_AD_CLOCK被设置为1，则采样率将除以二，此时标称采样率变为96、48、24及12千赫兹。

11.5 VS1053b Memory Map

X-memory		Y-memory		I-memory	
Address	**Description**	**Address**	**Description**	**Address**	**Description**
0x0000..0x17ff	System RAM	0x0000..0x17ff	System RAM	0x0000..0x004f	System RAM
0x1800..0x187f	User RAM	0x1800..0x187f	User RAM	0x0050..0x0fff	User RAM
0x1880..0x197f	Stack	0x1880..0x197f	Stack	0x1000..0x1fff	-
0x1980..0x3fff	System RAM	0x1980..0x3fff	System RAM	0x2000..0xffff	ROM 56k
0x4000..0xbfff	ROM 32k	0x4000..0xdfff	ROM 40k		and banked
0xc000..0xc0ff	Peripherals	0xe000..0xffff	System RAM	0xc000..0xffff	ROM4 16k
0xc100..0xffff	ROM 15.75k				

11.6 SCI Hardware Registers

SCI registers described in Chapter 9.6 can be found here between 0xC000..0xC00F. In addition to these registers, there is one in address 0xC010, called SCI_CHANGE.

SCI registers, prefix SCI_				
Reg	**Type**	**Reset**	**Abbrev[bits]**	**Description**
0xC010	r	0	CHANGE[5:0]	Last SCI access address

SCI_CHANGE bits		
Name	**Bits**	**Description**
SCI_CH_WRITE	4	1 if last access was a write cycle
SCI_CH_ADDR	3:0	SCI address of last access

SCI_CHANGE contains the last SCI register that has been accessed through the SCI bus, as well as whether the access was a read or write operation.

11.7 Serial Data Interface (SDI) Registers

Whenever two bytes have been written to the SDI bus, an interrupt is generated and the data can be read as a 16-bit big-endian value from the SDI registers. The user can control the DREQ pin as if it was a general-purpose output through its own register bit.

SDI registers, prefix SER_				
Reg	**Type**	**Reset**	**Abbrev[bits]**	**Description**
0xC011	r	0	DATA	Last received 2 bytes, big-endian
0xC012	w	0	DREQ[0]	DREQ pin control

11.5 VS1053b内存映射

X存储器	
地址	描述
0x0000..0x17ff	系统RAM
0x1800..0x187f	用户RAM
0x1880..0x197f	堆栈
0x1980..0x3fff	系统RAM
0x4000..0xbfff	ROM 32k
0xc000..0xc0ff	外设
0xc100..0xffff	ROM 15.75k

Y存储器	
地址	描述
0x0000..0x17ff	系统RAM
0x1800..0x187f	用户RAM
0x1880..0x197f	堆栈
0x1980..0x3fff	系统RAM
0x4000..0xdfff	ROM 40k
0xe000..0xffff	系统RAM

指令存储器	
地址	描述
0x0000..0x004f	系统RAM
0x0050..0x0fff	用户RAM
0x1000..0x1fff	-
0x2000..0xffff	ROM 56k
	及分库
0xc000..0xffff	ROM4 16k

11.6 SCI硬件寄存器

第九章第6节所述的SCI寄存器位于0xC000..0xC00F地址区间。此外，地址0xC010处存在名为SCI_CHANGE的寄存器

SCI寄存器，前缀SCI_				
寄存器	类型	复位值	缩写[位数]	描述
0xC010	r	0	CHANGE[5:0]	末次SCI访问地址

SCI_CHANGE寄存器位		
名称	位	描述
SCI_CH_WRITE	4	若末次操作为写入周期则置1
SCI_CH_ADDR	3:0	末次访问的SCI地址

SCI_CHANGE寄存器存储最后通过SCI总线访问的寄存器地址，并记录该访问是读操作还是写操作。

第11.7节 串行数据接口(SDI)寄存器

每当两个字节写入SDI总线时，将产生中断，此时可通过SDI寄存器以大端模式读取16位数据。用户可通过专用寄存器位控制DREQ引脚，使其作为通用输出引脚使用。

SDI寄存器（前缀SER_）				
寄存器	类型	复位值	缩写[位数]	描述
0xC011	r	0	数据	最后接收的2字节数据（大端模式）
0xC012	写入	0	DREQ[0]	DREQ引脚控制

11.8 DAC Registers

DAC registers, prefix DAC_				
Reg	**Type**	**Reset**	**Abbrev[bits]**	**Description**
0xC013	rw	0	FCTLL	DAC frequency control, 16 LSbs
0xC014	rw	0	FCTLH	DAC frequency control 4MSbs, PLL control
0xC015	rw	0	LEFT	DAC left channel PCM value
0xC016	rw	0	RIGHT	DAC right channel PCM value
0xC045	rw	0	VOL	DAC hardware volume

The internal 20-bit register DAC_FCTL is calculated from DAC_FCTLH and DAC_FCTLL registers as follows: DAC_FCTL = (DAC_FCTLH & 15) $\times$ 65536 + DAC_FCTLL. Highest supported value for DAC_FCTL is 0x80000.

If we define C = DAC_FCTL and X = XTALI in Hz, then the resulting samplerate f_s of the associated DAC SampleRate Converter is $f_s = C \times X \times 2^{-27}$.

Example:
If C = 0x80000 and X = 12.288 MHz then $f_s = 524288 \times (12.288 \times 10^6) \times 2^{-27} = 48000$ (Hz).

Note: FCTLH bits 13:4 are used for the PLL Controller. See Chapter 11.9 for details.

DAC_VOL bits		
Name	**Bits**	**Description**
LEFT_FINE	15:12	Left channel gain +0.0 dB. . .+5.5 dB (0 to 11)
LEFT_COARSE	11:8	Left channel attenuation in -6 dB steps
RIGHT_FINE	7:4	Right channel volume +0.0 dB. . .+5.5 dB (0 to 11)
RIGHT_COARSE	3:0	Right channel attenuation in -6 dB steps

Normally DAC_VOL is handled by the firmware. DAC_VOL depends on SCI_VOL and the bass and treble settings in SCI_BASS (and optionally SS_SWING bits in SCI_STATUS).

11.9 PLL Controller

The Phase-Locked Loop (PLL) controller is used to generate clock frequencies that are higher than the incoming (crystal-based) clock frequency. The PLL output is used by the CPU core and some peripherals.

Configurable features include:

- VCO Enable/Disable
- Select VCO or input clock to be output clock
- Route VCO frequency to output pin

第11.8节 DAC寄存器

DAC寄存器，前缀DAC_				
寄存器	类型	复位值	缩写[位数]	描述
0xC013	读写	0	FCTLL	DAC频率控制，低16位
0xC014	读写	0	FCTLH	DAC频率控制高4位，PLL控制
0xC015	读写	0	LEFT	DAC左声道PCM值
0xC016	读写	0	右声道	DAC右声道PCM值
0xC045	读写	0	VOL	DAC硬件音量

内部20位寄存器DAC_FCTL由DAC_FCTLH和DAC_FCTLL寄存器按以下公式计算：DAC_FCTL = (DAC_FCTLH & 15) ×65536 + DAC_FCTLL。DAC_FCTL最高支持值为0x80000。

若定义 C= DAC_FCTL且 X= XTALI（单位为赫兹），则关联DAC采样率转换器的输出采样率 f_s满足： $f_s = C \times X \times 2^{-27}$。

示例：
当 C= 0x80000且 X= 12.288兆赫时， $f_s = 524288 \times (12.288 \times 106) \times 2 - 27 = 48000$ (赫兹)。

注意：FCTLH寄存器的位[13:4]用于PLL控制器。详见第11.9章说明。

DAC_VOL寄存器位		
名称	位	描述
LEFT_FINE	15:12	左声道增益 +0.0 dB...+5.5 dB (0至11)
LEFT_COARSE	11:8	左声道衰减步进值（-6 dB/步）
RIGHT_FINE	7:4	右声道音量 +0.0 dB...+5.5 dB (0至11)
RIGHT_COARSE	3:0	右声道衰减步进值（-6 dB/步）

通常DAC_VOL由固件处理。DAC_VOL取值取决于SCI_VOL寄存器以及SCI_BASS寄存器中的高低音设置（可选地还取决于SCI_STATUS寄存器的SS_SWING位）。

11.9 锁相环(PLL)控制器

锁相环(PLL)控制器用于生成高于外部(基于晶振的)时钟频率的时钟信号。CPU内核及部分外设使用PLL输出时钟。

可配置功能包括：

- 压控振荡器(VCO)启用/禁用
- 选择VCO或输入时钟作为输出时钟
- 将VCO频率路由至输出引脚

- Select PLL clock multiplier

At the core of the PLL controller is the VCO, a high frequency oscillator, whose oscillation frequency is adjusted to be an integer multiple of some input frequency. As the name “Phase-Locked Loop” suggests, this is done by comparing the phase of the input frequency against the phase of a signal which is derived from the VCO output through frequency division.

If the system is stable, e.g. the comparison phase difference remains virtually zero, the PLL is said to be “in lock”. This means that the output frequency of the VCO is stable and reliable.

The PLL is preceded by a division-by-two unit. Thus, with a nominal XTALI = 12.288 MHz, the internal clock frequency CLKI can be adjusted with an accuracy of XTALI/2 = 6.144 MHz.

PLL control lies in DAC_FCTL bits 13:4. To see what bits 3:0 do, see Chapter 11.8.

FREQCTLH PLL bits, prefix FCH_		
Name	**Bits**	**Description**
PLL_LOCKED	13	0=lock failed since last test (read-only)
PLL_SET_LOCK	12	1:Sets FCH_PLL_LOCKED to 1 to start lock test
PLL_VCO_OUT_ENA	11	Route VCO to GPIO pin (VS1000:second cs pin)
PLL_FORCE_PLL	9	1:System clock is VCO / 0:System clock is inclk
PLL_DIV_INCLK	8	divide inclk by 2 (for 1.5, 2.5 or 3.5 x clk)
PLL_RATE	7:4	PLL rate control

The PLL locked status can be checked by generating a high-active pulse (writing first “1” , then “0”) to FCH_PLL_SET_LOCK and reading FCH_PLL_LOCKED. FCH_PLL_LOCKED is set to “1” along with the high level of FCH_PLL_SET_LOCK and to “0” whenever the PLL falls out of lock. So if the “1” remains in FCH_PLL_LOCKED, PLL is in sync.

The PLL controller’s operation is optimized for frequencies around 12. . . 13 MHz. If you use an 24. . . 26 MHz input clock, set the extra clock divider bit SM_CLK_RANGE in register SCI_MODE to 1 before activating the PLL.

It’s recommended to change the PLL rate in small steps and wait for the PLL to stabilize after each change. For diagnostic purposes, the PLL clock output (VCO) can be routed to an I/O pin so it can be scanned with an oscilloscope.

FCH_PLL_RATE (bits 7:4) control PLL multiplication rate. PLL multiplier is (FCH_PLL_RATE + 1). When FCH_PLL_RATE is 0, the VCO is powered down and output clock is forced to be input clock (same as if FCH_PLL_FORCE_PLL = 0).

- 选择PLL时钟倍频系数

PLL控制器的核心是压控振荡器(VCO)——一种高频振荡器，其振荡频率可调整为输入频率的整数倍。正如'锁相环'其名所示，该机制通过比较输入频率相位与经VCO输出分频后信号的相位来实现锁定。

若系统稳定（即比较相位差基本保持为零），则称PLL处于'锁定'状态。这意味着VCO的输出频率稳定可靠。

PLL前端配置了二分频单元。因此，在标称XTALI=12.288 MHz时，内部时钟频率CLKI的调节精度可达XTALI/2=6.144 MHz。

PLL控制由DAC_FCTL寄存器的位13:4实现。位域3:0的功能说明请参见第11.8章节

FREQCTLH锁相环控制位，前缀FCH_		
名称	**位**	**描述**
PLL_LOCKED	13	0=自上次测试后锁定失败（只读状态）
PLL_SET_LOCK	12	1：将FCH_PLL_LOCKED置1以启动锁定测试
PLL_VCO_OUT_ENA	11	将压控振荡器信号路由至GPIO引脚（VS1000机型：第二片选引脚）
PLL_FORCE_PLL	9	1：系统时钟源为VCO / 0：系统时钟源为输入时钟(inclk)
PLL_DIV_INCLK	8	输入时钟二分频（用于1.5/2.5/3.5倍时钟配置）
PLL_RATE	7:4	锁相环速率控制

可通过向FCH_PLL_SET_LOCK写入高有效脉冲（先写'1'再写'0'）并读取FCH_PLL_LOCKED来检测锁相环锁定状态。当FCH_PLL_SET_LOCK处于高电平时，FCH_PLL_LOCKED置'1'；当锁相环失锁时，该标志位自动清零。因此若“1”在FCH_PLL_LOCKED位中保持，则表明PLL处于同步状态。

PLL控制器的工作频率优化范围为12~13 MHz。若使用24~26 MHz输入时钟，需在激活PLL前将SCI_MODE寄存器中的额外时钟分频位SM_CLK_RANGE置1。

建议以小步长调整PLL速率，且每次更改后需等待PLL稳定。出于诊断目的，可将PLL时钟输出(VCO)路由至I/O引脚，以便通过示波器进行检测。

FCH_PLL_RATE（位7:4）控制PLL倍频速率。PLL倍频系数为(FCH_PLL_RATE + 1)。 当FCH_PLL_RATE为0时，VCO将被关闭，输出时钟强制等同于输入时钟（等同于FCH_PLL_FORCE_PLL=0时的状态）。

11.10 GPIO

GPIO registers, prefix GPIO_				
Reg	**Type**	**Reset**	**Abbrev[bits]**	**Description**
0xC017	rw	0	DDR[7:0]	Direction
0xC018	r	0	IDATA[11:0]	Values read from the pins
0xC019	rw	0	ODATA[7:0]	Values set to the pins

GPIO_DIR is used to set the direction of the GPIO pins. 1 means output. GPIO_ODATA remembers its values even if a GPIO_DIR bit is set to input.

GPIO_IDATA is used to read the pin states. In VS1053 also the SDI and SCI input pins can be read through GPIO_IDATA: SCLK = GPIO_IDATA[8], XCS = GPIO_IDATA[9], SI = GPIO_IDATA[10], and XDCS = GPIO_IDATA[11].

GPIO registers don't generate interrupts.

Note that in VS1053b the VSDSP registers can be read and written through the SCI_WRAMADDR and SCI_WRAM registers. You can thus use the GPIO pins quite conveniently.

11.10 GPIO

GPIO寄存器，前缀GPIO_				
寄存器	类型	复位值	缩写[位数]	描述
0xC017	读写	0	DDR[7:0]	方向
0xC018	r	0	IDATA[11:0]	从引脚读取的数值
0xC019	读写	0	ODATA[7:0]	设置到引脚的数值

GPIO_DIR寄存器用于设置GPIO引脚方向。1表示输出模式。即使GPIO_DIR位被设置为输入，GPIO_ODATA仍会保持其值。

GPIO_IDATA寄存器用于读取引脚状态。在VS1053中，SDI和SCI输入引脚状态也可通过GPIO_IDATA读取：SCLK = GPIO_IDATA[8], XCS = GPIO_IDATA[9], SI = GPIO_IDATA[10], XDCS = GPIO_IDATA[11]。

GPIO寄存器不会产生中断。

注意：在VS1053b中，可通过SCI_WRAMADDR和SCI_WRAM寄存器读写VSDSP寄存器。因此您可以相当方便地使用GPIO引脚。

11.11 Interrupt Control

Interrupt registers, prefix INT_				
Reg	**Type**	**Reset**	**Abbrev[bits]**	**Description**
0xC01A	rw	0	ENABLE[9:0]	Interrupt enable
0xC01B	w	0	GLOB_DIS[-]	Write to add to interrupt counter
0xC01C	w	0	GLOB_ENA[-]	Write to subtract from interrupt counter
0xC01D	rw	0	COUNTER[4:0]	Interrupt counter

INT_ENABLE controls the interrupts. The control bits are as follows:

INT_ENABLE bits		
Name	**Bits**	**Description**
INT_EN_SDM	9	Enable Sigma Delta Modulator interrupt
INT_EN_SRC	8	Enable SampleRate Converter interrupt
INT_EN_TIM1	7	Enable Timer 1 interrupt
INT_EN_TIM0	6	Enable Timer 0 interrupt
INT_EN_RX	5	Enable UART RX interrupt
INT_EN_TX	4	Enable UART TX interrupt
INT_EN_ADC	3	Enable AD modulator interrupt
INT_EN_SDI	2	Enable Data interrupt
INT_EN_SCI	1	Enable SCI interrupt
INT_EN_DAC	0	Enable DAC interrupt

Note: It may take upto 6 clock cycles before changing INT_ENABLE has any effect.

Writing any value to INT_GLOB_DIS adds one to the interrupt counter INT_COUNTER and effectively disables all interrupts. It may take upto 6 clock cycles before writing to this register has any effect.

Writing any value to INT_GLOB_ENA subtracts one from the interrupt counter INT_COUNTER, unless it already was 0, in which case nothing happens. If, after the operation INT_COUNTER becomes zero, interrupts selected with INT_ENABLE are restored. An interrupt routine should always write to this register as the last thing it does, because interrupts automatically add one to the interrupt counter, but subtracting it back to its initial value is the responsibility of the user. It may take upto 6 clock cycles before writing this register has any effect.

By reading INT_COUNTER the user may check if the interrupt counter is correct or not. If the register is not 0, interrupts are disabled.

11.11 中断控制

中断寄存器，前缀INT_				
寄存器	类型	复位值	缩写[位数]	描述
0xC01A	读写	0	ENABLE[9:0]	中断使能
0xC01B	写入	0	GLOB_DIS[-]	写入以增加中断计数器值
0xC01C	写入	0	GLOB_ENA[-]	写入以减少中断计数器值
0xC01D	读写	0	COUNTER[4:0]	中断计数器

INT_ENABLE寄存器控制中断功能控制位定义如下：

INT_ENABLE寄存器位		
名称	位	描述
INT_EN_SDM	9	启用Sigma Delta调制器中断
INT_EN_SRC	8	启用采样率转换器中断
INT_EN_TIM1	7	启用定时器1中断
INT_EN_TIM0	6	启用定时器0中断
INT_EN_RX	5	启用UART接收中断
INT_EN_TX	4	启用UART发送中断
INT_EN_ADC	3	启用AD调制器中断
INT_EN_SDI	2	启用数据中断
INT_EN_SCI	1	启用SCI中断
INT_EN_DAC	0	启用DAC中断

注意：修改INT_ENABLE后可能需要最多6个时钟周期才能生效。

向INT_GLOB_DIS写入任意值会使中断计数器INT_COUNTER增加1，并有效禁用所有中断。写入此寄存器后可能需要最多6个时钟周期才能生效。

向INT_GLOB_ENA写入任意值会使中断计数器INT_COUNTER减去1，除非其值已为0（此时无变化）。 若操作后INT_COUNTER变为零，则通过INT_ENABLE选定的中断将恢复启用。中断服务程序应始终在最后一步操作此寄存器，因为中断会自动使中断计数器加1，但将其减回初始值是用户的责任。

写入此寄存器后可能需要最多6个时钟周期才能生效。

通过读取 INT_COUNTER 寄存器，用户可验证中断计数器是否正常。若该寄存器值不为0，则表示中断功能已禁用。

11.12 UART

RS232 UART implements a serial interface using rs232 standard.

Figure 20: RS232 serial interface protocol

When the line is idling, it stays in logic high state. When a byte is transmitted, the transmission begins with a start bit (logic zero) and continues with data bits (LSB first) and ends up with a stop bit (logic high). 10 bits are sent for each 8-bit byte frame.

11.12.1 UART Registers

UART registers, prefix UART_				
Reg	**Type**	**Reset**	**Abbrev**	**Description**
0xC028	r	0	STATUS[4:0]	Status
0xC029	r/w	0	DATA[7:0]	Data
0xC02A	r/w	0	DATAH[15:8]	Data High
0xC02B	r/w	0	DIV	Divider

11.12.2 Status UART_STATUS

A read from the status register returns the transmitter and receiver states.

UART_STATUS Bits		
Name	**Bits**	**Description**
UART_ST_FRAMEERR	4	Framing error (stop bit was 0)
UART_ST_RXORUN	3	Receiver overrun
UART_ST_RXFULL	2	Receiver data register full
UART_ST_TXFULL	1	Transmitter data register full
UART_ST_TXRUNNING	0	Transmitter running

UART_ST_FRAMEERR is set if the stop bit of the received byte was 0.

UART_ST_RXORUN is set if a received byte overwrites unread data when it is transferred from the receiver shift register to the data register, otherwise it is cleared.

UART_ST_RXFULL is set if there is unread data in the data register.

UART_ST_TXFULL is set if a write to the data register is not allowed (data register full).

11.12 UART

RS232 UART 采用 rs232 标准实现串行接口。

图20：RS232 串行接口协议

线路空闲时保持逻辑高电平状态。字节传输时以起始位（逻辑0）开始，随后传输数据位（最低有效位优先），最后以停止位（逻辑高电平）结束。每个8位字节帧共发送10个比特。

11.12.1 UART寄存器

UART寄存器，前缀 UART_				
寄存器	类型	复位值	缩写	描述
0xC028	r	0	STATUS[4:0]	状态
0xC029	读写	0	DATA[7:0]	数据
0xC02A	读写	0	DATAH[15:8]	数据高位
0xC02B	读写	0	分频	分频器

11.12.2 状态寄存器 UART_STATUS

读取状态寄存器将返回发送器与接收器的状态信息。

UART_STATUS 寄存器位定义		
名称	位	描述
UART_ST_FRAMEERR	4	帧错误（停止位为0）
UART_ST_RXORUN	3	接收器溢流
UART_ST_RXFULL	2	接收数据寄存器满
UART_ST_TXFULL	1	发送数据寄存器满
UART_ST_TXRUNNING	0	发送器运行中

当接收字节的停止位为0时，UART_ST_FRAMEERR 位置位。

当接收字节从接收移位寄存器传输至数据寄存器时覆盖未读取数据，则 UART_ST_RXORUN 位置位，否则清零。

若数据寄存器中存在未读取数据，则 UART_ST_RXFULL 位置位。

当禁止写入数据寄存器时（数据寄存器满），UART_ST_TXFULL 位置位。

UART_ST_TXRUNNING is set if the transmitter shift register is in operation.

11.12.3 Data UART_DATA

A read from UART_DATA returns the received byte in bits 7:0, bits 15:8 are returned as '0'. If there is no more data to be read, the receiver data register full indicator will be cleared.

A receive interrupt will be generated when a byte is moved from the receiver shift register to the receiver data register.

A write to UART_DATA sets a byte for transmission. The data is taken from bits 7:0, other bits in the written value are ignored. If the transmitter is idle, the byte is immediately moved to the transmitter shift register, a transmit interrupt request is generated, and transmission is started. If the transmitter is busy, the UART_ST_TXFULL will be set and the byte remains in the transmitter data register until the previous byte has been sent and transmission can proceed.

11.12.4 Data High UART_DATAH

The same as UART_DATA, except that bits 15:8 are used.

11.12.5 Divider UART_DIV

UART_DIV Bits		
Name	**Bits**	**Description**
UART_DIV_D1	15:8	Divider 1 (0..255)
UART_DIV_D2	7:0	Divider 2 (6..255)

The divider is set to 0x0000 in reset. The ROM boot code must initialize it correctly depending on the master clock frequency to get the correct bit speed. The second divider (D_2) must be from 6 to 255.

The communication speed $f = \frac{f_m}{(D_1+1)\times(D_2)}$, where f_m is the master clock frequency, and f is the TX/RX speed in bps.

Divider values for common communication speeds at 26 MHz master clock:

若发送移位寄存器正在运行，则 UART_ST_TXRUNNING 位置位。

11.12.3 数据寄存器 UART_DATA

读取 UART_DATA 时，位7:0返回接收字节，位15:8固定返回'0'。 若无更多数据可读取，接收数据寄存器满指示器将清零。

当字节从接收移位寄存器移入接收数据寄存器时，将产生接收中断。

写入UART_DATA寄存器将设置待发送字节。数据取自位[7:0]，写入值中的其他位将被忽略。若发送器空闲，该字节将立即移入发送移位寄存器，产生发送中断请求，并启动传输过程。若发送器繁忙，则UART_ST_TXFULL标志位将被置位，该字节将保留在发送数据寄存器中，直至前一字节发送完毕方可继续传输。

11.12.4 数据高字节 UART_DATAH

功能与UART_DATA相同，但使用位[15:8]传输数据。

11.12.5 分频器 UART_DIV

UART_DIV 分频器位域		
名称	**位**	**描述**
UART_DIV_D1	15:8	分频系数1 (0至255)
UART_DIV_D2	7:0	分频系数2 (6至255)

复位时分频器值设为0x0000。 ROM引导代码必须根据主时钟频率正确初始化该寄存器，以获得精确的比特率。第二组分频系数(D_2)取值范围必须为6至255。

通信速度 $f = \frac{f_m}{(D_1+1)\times(D_2)}$，其中fm为主时钟频率，f为TX/RX传输速率（单位：bps）

主时钟26 MHz时常见通信速率的分频器设置值：

Example UART Speeds, $f_m = 49.152\,MHz$		
Comm. Speed [bps]	UART_DIV_D1	UART_DIV_D2
4800	255	40
9600	255	20
14400	233	15
19200	255	10
28800	243	7
38400	159	8
57600	121	7
115200	60	7

11.12.6 UART Interrupts and Operation

Transmitter operates as follows: After an 8-bit word is written to the transmit data register it will be transmitted instantly if the transmitter is not busy transmitting the previous byte. When the transmission begins a TX_INTR interrupt will be sent. Status bit [1] informs the transmitter data register empty (or full state) and bit [0] informs the transmitter (shift register) empty state. A new word must not be written to transmitter data register if it is not empty (bit [1] = '0'). The transmitter data register will be empty as soon as it is shifted to transmitter and the transmission is begun. It is safe to write a new word to transmitter data register every time a transmit interrupt is generated.

Receiver operates as follows: It samples the RX signal line and if it detects a high to low transition, a start bit is found. After this it samples each 8 bit at the middle of the bit time (using a constant timer), and fills the receiver (shift register) LSB first. Finally the data in the receiver is moved to the reveive data register, the stop bit state is checked (logic high = ok, logic low = framing error) for status bit[4], the RX_INTR interrupt is sent, status bit[2] (receive data register full) is set, and status bit[2] old state is copied to bit[3] (receive data overrun). After that the receiver returns to idle state to wait for a new start bit. Status bit[2] is zeroed when the receiver data register is read.

RS232 communication speed is set using two clock dividers. The base clock is the processor master clock. Bits 15-8 in these registers are for first divider and bits 7-0 for second divider. RX sample frequency is the clock frequency that is input for the second divider.

UART速率示例， $f_m = 49.152$兆赫兹		
通信速率 [bps]	UART_DIV_D1	UART_DIV_D2
4800	255	40
9600	255	20
14400	233	15
19200	255	10
28800	243	7
38400	159	8
57600	121	7
115200	60	7

11.12.6 UART中断与操作

发送器工作流程：当8位数据写入发送数据寄存器后，若发送器未处于前一字节的发送状态，将立即启动传输传输开始时将触发TX_INTR中断状态位[1]指示发送数据寄存器空/满状态，位[0]指示发送器（移位寄存器）空状态当发送数据寄存器非空时（位[1] = '0'），禁止写入新数据数据从发送寄存器移入发送器启动传输后，发送数据寄存器将立即变为空状态每次产生发送中断时向发送数据寄存器写入新字都是安全的。

接收器工作流程如下：采样RX信号线，若检测到高电平到低电平的跳变，则判定找到起始位。此后在比特时间中点采样每个8位数据（使用恒定定时器），并以最低有效位优先的方式填充接收器（移位寄存器）。 最终将接收器数据移入接收数据寄存器，检测停止位状态（逻辑高=正常，逻辑低=帧错误）并更新状态位[4]，发送RX_INTR中断，置位状态位[2]（接收数据寄存器满），同时将状态位[2]的旧状态复制到位[3]（接收数据溢出）。 随后接收器返回空闲状态等待新起始位。读取接收数据寄存器时状态位[2]将被清零。

RS232通信速率通过两个时钟分频器设置。基准时钟即处理器主时钟。这些寄存器中位15-8用于第一分频器，位7-0用于第二分频器。接收采样频率是输入至第二分频器的时钟频率。

11.13 Timers

There are two 32-bit timers that can be initialized and enabled independently of each other. If enabled, a timer initializes to its start value, written by a processor, and starts decrementing every clock cycle. When the value goes past zero, an interrupt is sent, and the timer initializes to the value in its start value register, and continues downcounting. A timer stays in that loop as long as it is enabled.

A timer has a 32-bit timer register for down counting and a 32-bit TIMER1_LH register for holding the timer start value written by the processor. Timers have also a 2-bit TIMER_ENA register. Each timer is enabled (1) or disabled (0) by a corresponding bit of the enable register.

11.13.1 Timer Registers

Timer registers, prefix TIMER_				
Reg	**Type**	**Reset**	**Abbrev**	**Description**
0xC030	r/w	0	CONFIG[7:0]	Timer configuration
0xC031	r/w	0	ENABLE[1:0]	Timer enable
0xC034	r/w	0	T0L	Timer0 startvalue - LSBs
0xC035	r/w	0	T0H	Timer0 startvalue - MSBs
0xC036	r/w	0	T0CNTL	Timer0 counter - LSBs
0xC037	r/w	0	T0CNTH	Timer0 counter - MSBs
0xC038	r/w	0	T1L	Timer1 startvalue - LSBs
0xC039	r/w	0	T1H	Timer1 startvalue - MSBs
0xC03A	r/w	0	T1CNTL	Timer1 counter - LSBs
0xC03B	r/w	0	T1CNTH	Timer1 counter - MSBs

11.13.2 Configuration TIMER_CONFIG

TIMER_CONFIG Bits		
Name	**Bits**	**Description**
TIMER_CF_CLKDIV	7:0	Master clock divider

TIMER_CF_CLKDIV is the master clock divider for all timer clocks. The generated internal clock frequency $f_i = \frac{f_m}{c+1}$, where f_m is the master clock frequency and c is TIMER_CF_CLKDIV. Example: With a 12 MHz master clock, TIMER_CF_DIV=3 divides the master clock by 4, and the output/sampling clock would thus be $f_i = \frac{12MHz}{3+1} = 3MHz$.

11.13 定时器

存在两个可独立初始化和使能的32位定时器。若使能，定时器将初始化至处理器写入的起始值，并在每个时钟周期递减计数。当计数值经过零时触发中断，定时器将重新加载起始值寄存器中的数值并继续递减计数。只要定时器处于使能状态，就会保持该循环工作模式。

每个定时器拥有一个用于递减计数的32位定时器寄存器，以及一个存储处理器写入的起始值的32位TIMER1_LH寄存器。定时器还包含一个2位的TIMER_ENA寄存器。每个定时器的使能状态(1)或禁用状态(0)由使能寄存器的对应位控制。

11.13.1 定时器寄存器

定时器寄存器，前缀 TIMER_				
寄存器	类型	复位值	缩写	描述
0xC030	读写	0	CONFIG[7:0]	定时器配置
0xC031	读写	0	ENABLE[1:0]	定时器使能
0xC034	读写	0	T0L	定时器0起始值 - 低位字节
0xC035	读写	0	T0H	定时器0起始值 - 高位字节
0xC036	读写	0	T0CNTL	定时器0计数器 - 低位字节
0xC037	读写	0	T0CNTH	定时器0计数器 - 最高有效位
0xC038	读写	0	T1L	定时器1起始值 - 最低有效位
0xC039	读写	0	T1H	定时器1起始值 - 最高有效位
0xC03A	读写	0	T1CNTL	定时器1计数器 - 最低有效位
0xC03B	读写	0	T1CNTH	定时器1计数器 - 最高有效位

11.13.2 配置寄存器 TIMER_CONFIG

TIMER_CONFIG 位域		
名称	位	描述
TIMER_CF_CLKDIV	7:0	主时钟分频器

TIMER_CF_CLKDIV 是所有定时器时钟的主时钟分频系数。生成的内部时钟频率 $f\_i = f\_m / (c+1)$，其中 f_m为主时钟频率， c为 TIMER_CF_CLKDIV 的值。
示例：当主时钟为12 MHz时，TIMER_CF_DIV=3会将主时钟进行4分频，因此输出/采样时钟频率为 $f_i = \frac{12\text{兆赫兹}}{3+1} = 3$ 兆赫兹。

11.13.3 Configuration TIMER_ENABLE

TIMER_ENABLE Bits		
Name	**Bits**	**Description**
TIMER_EN_T1	1	Enable timer 1
TIMER_EN_T0	0	Enable timer 0

11.13.4 Timer X Startvalue TIMER_Tx[L/H]

The 32-bit start value TIMER_Tx[L/H] sets the initial counter value when the timer is reset. The timer interrupt frequency $f_t = \frac{f_i}{c+1}$ where f_i is the master clock obtained with the clock divider (see Chapter 11.13.2 and c is TIMER_Tx[L/H].

Example: With a 12 MHz master clock and with TIMER_CF_CLKDIV=3, the master clock $f_i = 3MHz$. If TIMER_TH=0, TIMER_TL=99, then the timer interrupt frequency $f_t = \frac{3MHz}{99+1} = 30kHz$.

11.13.5 Timer X Counter TIMER_TxCNT[L/H]

TIMER_TxCNT[L/H] contains the current counter values. By reading this register pair, the user may get knowledge of how long it will take before the next timer interrupt. Also, by writing to this register, a one-shot different length timer interrupt delay may be realized.

11.13.6 Timer Interrupts

Each timer has its own interrupt, which is asserted when the timer counter underflows.

11.13.3 使能寄存器 TIMER_ENABLE

定时器使能位		
名称	**位**	**描述**
TIMER_EN_T1	1	使能定时器1
TIMER_EN_T0	0	使能定时器0

11.13.4 定时器X起始值 TIMER_Tx[L/H]

32位起始值TIMER_Tx[L/H]用于设置定时器复位时的计数器初始值。定时器中断频率 $f_t = \frac{f_i}{c+1}$ 其中 f_i 是通过时钟分频器获得的主时钟频率（参见第11.13.2章）， c 为TIMER_Tx[L/H]的值。

示例：当主时钟为12 MHz且TIMER_CF_CLKDIV=3时，主时钟频率 $f_i = 3$ 兆赫兹。 若TIMER_TH=0，TIMER_TL=99，则定时器中断频率 $f_t = \frac{3\text{兆赫兹}}{99+1} =$ 99+1= 100，因此30千赫兹。

11.13.5 定时器X计数器 TIMER_TxCNT[L/H]（高低位）

TIMER_TxCNT[L/H] 寄存器包含当前计数器值。通过读取该寄存器对，用户可获知下一次定时器中断发生前的剩余时间。同时，通过写入此寄存器可实现不同延时长度的一次性定时器中断。

11.13.6 定时器中断

每个定时器拥有独立的中断信号，当定时器计数器下溢时触发中断。

11.14 I2S DAC Interface

The I2S Interface makes it possible to attach an external DAC to the system.

Note: The samplerate of the audio file and the I2S rate are independent. All audio will be automatically converted to 6.144 MHz for VS1053 DAC and to the configured I2S rate using a high-quality sample-rate converter.

Note: In VS1053b the I2S pins share different GPIO pins than in VS1033 to be able to use SPI boot and I2S in the same application.

I2S registers, prefix I2S_				
Reg	**Type**	**Reset**	**Abbrev**	**Description**
0xC040	r/w	0	CONFIG[3:0]	I2S configuration

I2S_CONFIG Bits		
Name	**Bits**	**Description**
I2S_CF_MCLK_ENA	3	Enables the MCLK output (12.288 MHz)
I2S_CF_ENA	2	Enables I2S, otherwise pins are GPIO
I2S_CF_SRATE	1:0	I2S rate, "10" = 192, "01" = 96, "00" = 48 kHz

I2S_CF_ENA enables the I2S interface. After reset I2S is disabled and the pins are used for GPIO inputs.

I2S_CF_MCLK_ENA enables the MCLK output. The frequency is either directly the input clock (nominal 12.288 MHz), or half the input clock when mode register bit SM_CLK_RANGE is set to 1 (24-26 MHz input clock).

I2S_CF_SRATE controls the output samplerate. When set to 48 kHz, SCLK is MCLK divided by 8, when 96 kHz SCLK is MCLK divided by 4, and when 192 kHz SCLK is MCLK divided by 2. I2S_CF_SRATE can only be changed when I2S_CF_ENA is 0.

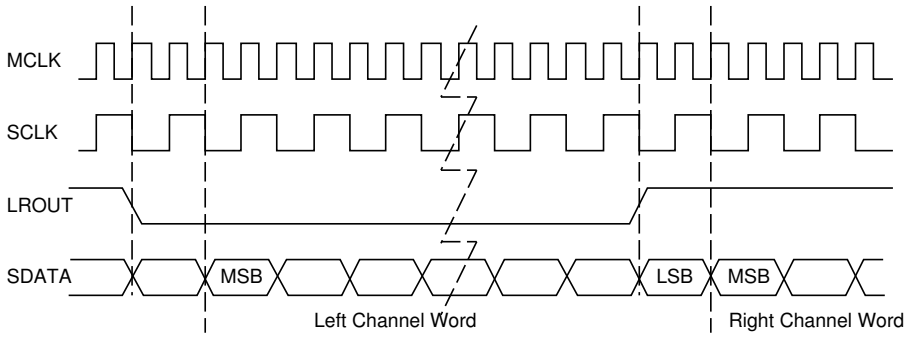


Figure 21: I2S interface, 192 kHz.

To enable I2S first write 0xc017 to SCI_WRAMADDR and 0xf0 to SCI_WRAM, then write 0xc040 to SCI_WRAMADDR and 0x0c to SCI_WRAM.

11.14 I2S数模转换器接口

I2S接口支持连接外部DAC至本系统。

注：音频文件的采样率与I2S速率相互独立。所有音频将通过高质量采样率转换器自动转换为6.144 MHz（用于VS1053 DAC）或配置的I2S速率。

注：VS1053b的I2S引脚与VS1033共享不同的GPIO引脚，以便在同一应用中兼容SPI启动和I2S功能。

I2S寄存器 (前缀 I2S_)				
寄存器	类型	复位值	缩写	描述
0xC040	读写	0	CONFIG[3:0]	I2S配置

I2S_CONFIG位域		
名称	位	描述
I2S_CF_MCLK_ENA	3	使能MCLK输出（12.288 MHz）
I2S_CF_ENA	2	启用I2S功能，否则引脚用作GPIO
I2S_CF_SRATE	1:0	I2S速率："10"=192，"01"=96，"00"=48 kHz

I2S_CF_ENA位使能I2S接口复位后I2S处于禁用状态，相关引脚用作GPIO输入

I2S_CF_MCLK_ENA位使能MCLK输出输出频率为直接输入时钟（标称值12.288 MHz），或当模式寄存器位SM_CLK_RANGE设为1时（输入时钟24-26 MHz）为输入时钟的一半

I2S_CF_SRATE位控制输出采样率设为48kHz时，SCLK为MCLK除以8；设为96kHz时，SCLK为MCLK除以4；设为192kHz时，SCLK为MCLK除以2仅当I2S_CF_ENA为0时方可修改I2S_CF_SRATE。

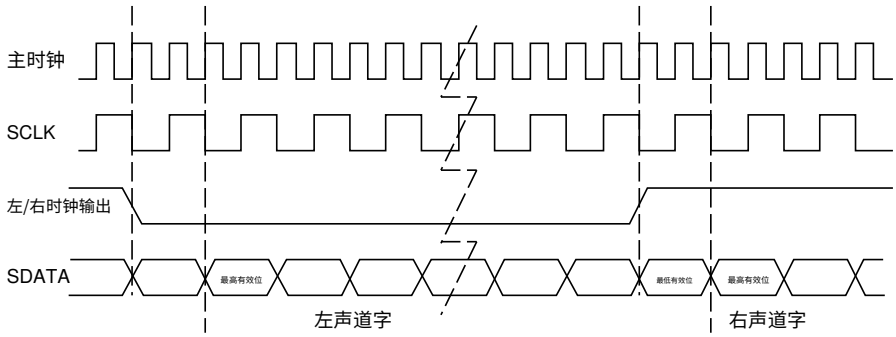


图21：I2S接口工作示意图（192 kHz）

启用I2S需先向SCI_WRAMADDR写入0xc017并向SCI_WRAM写入0xf0，随后向SCI_WRAMADDR写入0xc040并向SCI_WRAM写入0x0c。

11.15 Analog-to-Digital Converter (ADC)

ADC modulator registers control Analog-to-Digital conversions of VS1053b.

ADC Decimator registers, prefix ADC_				
Reg	**Type**	**Reset**	**Abbrev[bits]**	**Description**
0xC042	rw	0	CONTROL[4:0]	ADC control
0xC043	r	0	DATA_LEFT	ADC left channel data
0xC044	r	0	DATA_RIGHT	ADC right channel data

ADC_CONTROL controls the ADC and its associated decimator unit.

ADC_CONTROL Bits		
Name	**Bits**	**Description**
ADC_MODU2_PD	4	Right channel powerdown
ADC_MODU1_PD	3	Left channel powerdown
ADC_DECIM_FACTOR	2:1	ADC Decimator factor: - 3 = downsample to XTALI/512 (nominal 24 kHz) - 2 = downsample to XTALI/256 (nominal 48 kHz) - 1 = downsample to XTALI/128 (nominal 96 kHz) - 0 = downsample to XTALI/64 (nominal 192 kHz)
ADC_ENABLE	0	Set to activate ADC converter and decimator

Note: Setting bit SS_AD_CLOCK in register SCI_STATUS will halve the operation speed of the A/D unit, and thus halve the resulting samplerate.

Each time a new (stereo) sample has been generated, an ADC interrupt is generated.

11.15 模数转换器 (ADC)

ADC调制器寄存器控制VS1053b的模数转换功能。

ADC抽取滤波器寄存器（前缀ADC_）				
寄存器	类型	复位值	缩写[位数]	描述
0xC042	读写	0	CONTROL[4:0]	ADC控制
0xC043	r	0	DATA_LEFT	ADC左声道数据
0xC044	r	0	DATA_RIGHT	ADC右声道数据

ADC_CONTROL寄存器控制ADC及其关联的抽取器单元。

ADC_CONTROL寄存器位定义		
名称	位	描述
ADC_MODU2_PD	4	右声道断电
ADC_MODU1_PD	3	左声道断电
ADC_DECIM_FACTOR	2:1	ADC抽取因子： - 3 = 下采样至XTALI/512（标称24 kHz） - 2 = 下采样至XTALI/256（标称48 kHz） - 1 = 下采样至XTALI/128（标称96 kHz） - 0 = 下采样至XTALI/64（标称192 kHz）
ADC_ENABLE	0	置位以激活ADC转换器及抽取器

注意：设置SCI_STATUS寄存器的SS_AD_CLOCK位将使A/D单元运行速度减半，从而导致采样率减半。

每次生成新（立体声）采样时，会产生ADC中断。

11.16 Resampler SampleRate Converter (SRC)

The resampler SRC makes it possible to catch audio from the DAC path.

Note: hardware makes no attempts at low-pass filtering data. If the SRC samplerate is lower than the DAC samplerate, aliasing may and will occur.

Resampler SRC registers, prefix SRC_				
Reg	**Type**	**Reset**	**Abbrev[bits]**	**Description**
0xC046	rw	0	CONTROL[12:0]	SRC control
0xC047	r	0	DATA_LEFT	SRC left channel data
0xC048	r	0	DATA_RIGHT	SRC right channel data

SRC_CONTROL Bits		
Name	**Bits**	**Description**
SRC_ENABLE	12	Set to enable SRC
SRC_DIV	11:0	Set samplerate to XTALI/2/(SRC_DIV+1)

Each time a new (stereo) sample has been generated, an SRC interrupt is generated.

11.16 重采样器采样率转换器 (SRC)

重采样器SRC可实现从DAC路径捕获音频。

注意：硬件不会对数据进行低通滤波处理。若SRC采样率低于DAC采样率，则混叠效应可能并且将会发生。

重采样器SRC寄存器，前缀SRC_				
寄存器	类型	复位值	缩写[位数]	描述
0xC046	读写	0	CONTROL[12:0]	SRC控制寄存器
0xC047	r	0	DATA_LEFT	SRC左声道数据
0xC048	r	0	DATA_RIGHT	SRC右声道数据

SRC_CONTROL寄存器位定义		
名称	位	描述
SRC_ENABLE	12	置位使能SRC功能
SRC_DIV	11:0	设置采样率为XTALI/2/(SRC_DIV+1)

每次生成新立体声样本时，将产生SRC中断

11.17 Sidestream Sigma-Delta Modulator (SDM)

The Sidestream Sigma-Delta Modulator makes it possible to insert a digital side stream on top of existing audio.

Note: The SDM provides a direct, low-delay side channel to the Sigma-Delta DACs of VS10xx. It makes no attempts at low-pass filtering data. Thus there will be practically no image rejection. If using low samplerates, this may cause audible aliasing distortion.

Sidestream SDM registers, prefix SDM_				
Reg	**Type**	**Reset**	**Abbrev[bits]**	**Description**
0xC049	rw	0	CONTROL[12:0]	SDM control
0xC04A	rw	0	DATA_LEFT	SDM left channel data
0xC04B	rw	0	DATA_RIGHT	SDM right channel data

SDM_CONTROL Bits		
Name	**Bits**	**Description**
SDM_ENABLE	12	Set to enable SDM
SDM_DIV	11:0	Set samplerate to XTALI/2/(SDM_DIV+1)

Each time a new (stereo) sample is needed, an SDM interrupt is generated.

11.17 旁路式Sigma-Delta调制器 (SDM)

旁路Σ-Δ调制器可在现有音频信号上叠加数字旁路流

注：SDM为VS10xx的Σ-Δ数模转换器提供直接低延迟旁路通道
该模块不进行低通滤波处理因此实际上不具备镜像抑制功能
若使用低采样率，可能导致可闻的混叠失真

旁路SDM寄存器，前缀SDM_				
寄存器	类型	复位值	缩写[位数]	描述
0xC049	读写	0	CONTROL[12:0]	SDM 控制寄存器
0xC04A	读写	0	DATA_LEFT	SDM 左声道数据
0xC04B	读写	0	DATA_RIGHT	SDM 右声道数据

SDM_CONTROL 寄存器位定义		
名称	位	描述
SDM_ENABLE	12	置位使能 SDM 功能
SDM_DIV	11:0	设置采样率为 XTALI/2/(SDM_DIV+1)

每次需要新（立体声）样本时，将产生 SDM 中断

12 Version Changes

This chapter describes the lastest and most important changes done to VS1053b

12.1 Changes Between VS1033c and VS1053a/b Firmware, 2007-03-08

Completely new or major changes:

- I2S pins are now in GPIO4-GPIO7 and do not overlap with SPI boot pins.
- No software reset required between files when used correctly.
- Ogg Vorbis decoding added. Non-fatal ogg or vorbis decode errors cause automatic resync. This allows easy rewind and fast forward. Decoding ends if the "last frame" flag is reached or SM_CANCEL is set.
- HE-AAC v2 Level 3 decoding added. It is possible to disable PS and SBR processing and control the upsampling modes through `parametric_x.control1`.
- Like the WMA decoder, the AAC decoder uses the clock adder (see SCI_CLOCKF) if it needs more clock to decode the file. HE-AAC features are dropped one by one, if the file can not be decoded correctly even with the highest allowed clock. Parametric stereo is the first feature to be dropped, then downsampled mode is used, and as the final resort Spectral Band Replication is disabled. Features are automatically restored for the next file.
- Completely new volume control with zero-cross detection prevents pops when volume is changed.
- Audio FIFO underrun detection (with slow fade to zero) instead of looping the audio buffer content.
- Average bitrate calculation (`byteRate`) for all codecs.
- All codecs support fast play mode with selectable speeds for the best-quality fast forward operation. Fast play also advances DECODE_TIME faster.
- WMA and Ogg Vorbis provide an absolute decode position in milliseconds.
- When SM_CANCEL is detected, the firmware also discards the stream buffer contents.
- Bit SCIST_DO_NOT_JUMP in SCI_STATUS is ’1’ when jumps in the file should not be done: during header processing and with Midi files.
- IMA ADPCM encode now supports stereo encoding and selectable samplerate.

Other changes or additions:

- Delayed volume and bass/treble control calculation reduces the time the corresponding SCI operations take. This delayed handling and the new volume control hardware prevents audio samples from being missed during volume change.
- SCI_DECODE_TIME only cleared at hardware and software reset to allow files to be played back-to-back or looped.

12 版本变更记录

本章描述 VS1053b 的最新重要变更

12.1 VS1033c 与 VS1053a/b 固件变更对比 (2007-03-08)

全新或重大变更项：

- I2S 引脚现位于 GPIO4-GPIO7，与 SPI 启动引脚无冲突
- 正确使用时，文件切换无需软件复位
- 新增 Ogg Vorbis 格式解码功能非致命性 Ogg 或 Vorbis 解码错误将触发自动重新同步该功能支持便捷的倒带和快进操作。若达到"最后一帧"标志或SM_CANCEL寄存器被置位，则解码终止。
- 新增HE-AAC v2 Level 3解码支持。可通过parametric_x.control1禁用参数立体声(PS)和频带复制(SBR)处理，并控制上采样模式。
- 与WMA解码器类似，当AAC解码器需要更高时钟频率解码文件时，会启用时钟加法器功能（参见SCI_CLOCKF寄存器）。 若即使在允许的最高时钟频率下仍无法正确解码文件，HE-AAC功能特性将逐级舍弃。参数立体声为首要舍弃项，其次启用降采样模式，最终手段为禁用频谱带复制(SBR)功能。后续文件将自动恢复所有功能特性。

- 全新音量控制系统采用过零检测技术，可消除音量调节时的爆破音。

- 新增音频FIFO欠载检测机制（采用缓降归零技术），替代原有音频缓冲区内容循环播放方案。
- 所有编解码器的平均比特率计算（byteRate）。
- 所有编解码器均支持快速播放模式，可选择不同速度以实现最佳质量快进操作。快速播放模式同时会加快DECODE_TIME的推进速度。
- WMA和Ogg Vorbis格式提供以毫秒为单位的绝对解码位置信息。
- 当检测到SM_CANCEL信号时，固件会同时丢弃流缓冲区内容。
- 当不应执行文件跳转时（如文件头处理期间及MIDI文件播放时），SCI_STATUS寄存器中的SCIST_DO_NOT_JUMP位将置为'1'。
- IMA ADPCM编码现支持立体声编码及可调采样率。

其他变更或新增功能：

- 延迟的音量及低音/高音控制计算缩短了相应SCI操作的执行时间。该延迟处理机制配合新型音量控制硬件，可防止音量调整期间丢失音频采样数据。

- SCI_DECODE_TIME仅在硬件与软件复位时清零，以实现文件连续播放或循环播放。

- Read and write to YRAM at 0xe000..0xffff added to SCI_WRAMADDR/SCI_WRAM.
- The `resync` parameter (parametric_x.resync) is set to 32767 after reset to allow inifinite resynchronization attempts (or until SM_CANCEL is set). Old operation can be restored by writing 0 to `resync` after reset.
- WMA,AAC: more robust resync.
- WMA,AAC: If resync is performed, broadcast mode is automatically activated. The broadcast mode disables file size checking, and decoding continues until SM_CANCEL is set or reset is performed.
- Treble control fixed (volume change could cause bad artefacts).
- MPEG Layer I mono fixed.
- MPEG Layer II half-rate decoding fixed (frame size was calculated wrong).
- MPEG Layer II accuracy problem fixed, invalid grouped values set to 0.
- WAV parser now skips unknown RIFF chunks.
- IMA ADPCM: Maximum blocksize is now 4096 bytes (4088 samples stereo, 8184 mono). Thus, now also plays 44100Hz stereo.
- Rt-midi: starts if in reset GPIO0='0', GPIO1='1', GPIO2&3 give earSpeaker setup.
- NewSinTest() and NewSinSweep() added (AIADDR = 0x4020/0x4022) AICTRL0 and AICTRL1 set sin frequency for left/right.
- Clears memory before SPI boot and not in InitHardware().

Known quirks, bugs, or features in VS1053b:

- Setting volume clears SS_REFERENCE_SEL and SS_AD_CLOCK bits. See Chapter 9.6.2.
- Software reset clears GPIO_DDR, also affects I2S pins.
- Ogg Vorbis occasionally overflows in windowing causing a small glitch to audio. Patch available (*VS1053b Patches w/ FLAC Decoder* plugin at *http://www.vlsi.fi/en/support/software/vs10xxplugins.html*).
- IMA ADPCM encoding requires short patch to start. Patch available in Chapter 10.8.1.
- There are also fixes for some other issues, we recommend you use the latest version of the *VS1053b Patches w/ FLAC Decoder* package from *http://www.vlsi.fi/en/support/software/vs10xxplugins.html*.

- 新增通过SCI_WRAMADDR/SCI_WRAM对YRAM区域(0xe000..0xffff)的读写功能。
- 复位后，重新同步参数（parametric_x.resync）被设置为32767，以实现无限次重新同步尝试（直至SM_CANCEL被置位）。 复位后向重新同步参数写入0可恢复原始操作模式。

- WMA/AAC：增强型重新同步容错机制。
- WMA/AAC：若执行重新同步，广播模式将自动激活。广播模式会禁用文件大小校验，解码将持续进行直至SM_CANCEL置位或执行复位操作。

- 高音控制功能已修复（音量变化可能导致伪影失真问题已解决）。
- MPEG Layer I 单声道解码问题已修复。
- MPEG Layer II 半速率解码已修正（帧尺寸计算错误问题）。
- MPEG Layer II 精度问题已修复，无效分组值现归零处理。
- WAV解析器现已跳过未知RIFF数据块。
- IMA ADPCM：最大块尺寸现为4096字节（立体声4088样本，单声道8184样本）。因此，现可额外支持播放44100赫兹立体声音频。
- 实时MIDI功能启动条件：复位状态下GPIO0='0'、GPIO1='1'时，GPIO2与GPIO3配置耳机/扬声器输出模式。
- 新增NewSinTest()及NewSinSweep()函数（AIADDR = 0x4020/0x4022），通过AICTRL0与AICTRL1寄存器分别设置左右声道正弦波频率。
- SPI启动前执行内存清除操作，该操作不包含在InitHardware()函数中。

VS1053b已知特性、缺陷或注意事项：

- 音量设置会清除SS_REFERENCE_SEL与SS_AD_CLOCK寄存器位。详见章节9.6.2。

- 软件复位将清除GPIO_DDR寄存器配置，同时影响I2S引脚状态。
- Ogg Vorbis格式在加窗处理时偶发数据溢出，导致音频出现轻微杂音。补丁文件已发布（含*FLAC*解码器的*VS1053b*补丁插件详见*http://www.vlsi.fi/en/support/software/vs10xxplugins.html*）。
- IMA ADPCM编码需要短补丁启动。补丁详见第10.8.1章节。
- 同时包含若干其他问题的修复，建议您使用最新版

  *VS1053b*补丁含*FLAC*解码器软件包，下载地址：*http://www.vlsi.fi/en/support/software/vs10xxplugins.html*。

13 Document Version Changes

This chapter describes the most important changes to this document.

Version 1.20, 2012-12-03

- Major update to Chapter 11 *VS1053b Registers*. Added Chapter 11.3, *VS1053b Hardware DAC Audio Paths*, Chapter 11.4, *VS1053b Hardware ADC Audio Paths*, Chapter 11.9, *PLL Controller*, Chapter 11.15, *Analog-to-Digital Converter (ADC)*, Chapter 11.16, *Resampler SampleRate Converter (SRC)*, and Chapter 11.17, *Sidestream Sigma-Delta Modulator (SDM)*. Also revised several other sections.
- Fixed SCI_MODE default value in Chapter 9.6.1, *SCI_MODE (RW)*.
- Added info on how to read DREQ through SCI to Chapter 7.2, *Data Request Pin DREQ*.
- Slight reorganization to make datasheet more similar to VS1063a Datasheet.

Version 1.13, 2011-05-27

- xRESET, XTALI and XTALO high-level are referenced from IOVDD in Chapter 4.5.

Version 1.12, 2010-10-28

- Fixed the real-time MIDI through SDI documentation.

Version 1.11, 2010-04-30

- Minor updates.

Version 1.10, 2009-09-04

- Added mentions of new Ogg Vorbis encoder and FLAC decoder plugins.
- PCM recording documentation enhanced (Chapters 10.8 and 10.8.4).
- SCLK, XCS, SI, XDCS can be read through GPIO_IDATA.
- I2S rate and audio rate are independent.

13 文档版本修订记录

本章描述本文档的重要更新内容。

版本1.20，2012年12月3日

- 第11章*VS1053b*寄存器重大更新新增第11.*3*章*VS1053b*硬件*D*AC音频路径、第11.4章*VS1053b*硬件A*DC*音频路径、第11.9章*PLL*控制器、第11.1*5*章模数转换器*(ADC)*、第11.16章重采样器采样率转换器*(SRC)*以及第11.17章旁路*Sigma-Delta*调制器*(SDM)*。 同时修订了其他若干章节内容。

- 修正了第9.6.1章*SCI_MODE (RW)*中的SCI_MODE寄存器默认值。
- 在第7.2章数据请求引脚*DREQ*中新增通过SCI读取DREQ状态的方法说明。
- 小幅重组内容使数据手册格式更接近VS1063a数据手册。

版本1.13，2011-05-27

- 第4.5章中明确xRESET、XTALI和XTALO的高电平参考基准为IOVDD。

版本1.12，2010-10-28

- 修正了通过SDI实现实时MIDI的文档说明。

版本1.11，2010-04-30

- 细微更新。

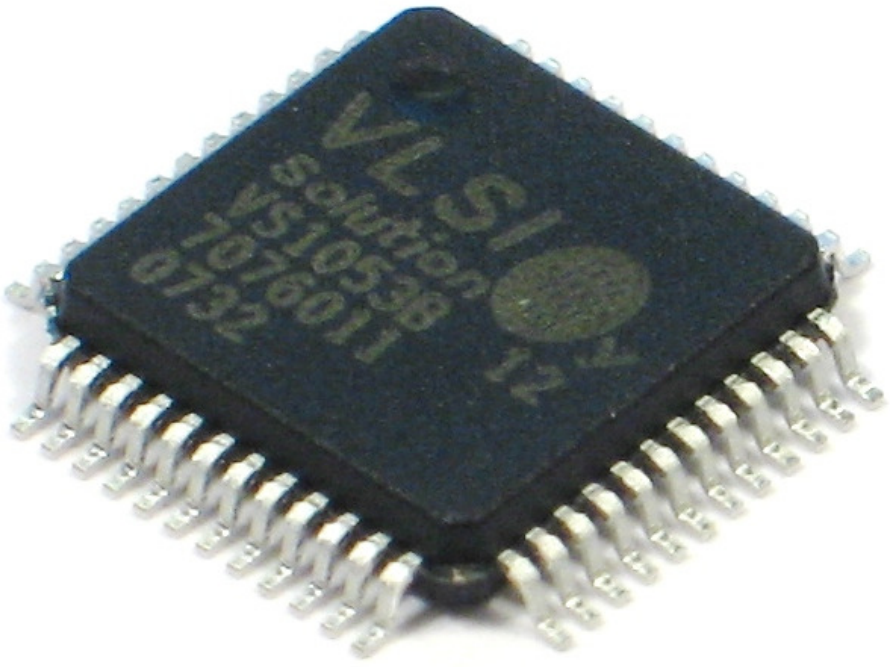
版本1.10，2009-09-04

- 新增对全新Ogg Vorbis编码器和FLAC解码器插件的说明。
- 增强PCM录音功能文档（第10.8章及10.8.4节）。
- 可通过GPIO_IDATA读取SCLK、XCS、SI、XDCS引脚状态。
- I2S速率与音频速率相互独立

14 Contact Information

VLSI Solution Oy
Entrance G, 2nd floor
Hermiankatu 8
FI-33720 Tampere
FINLAND

Fax: +358-3-3140-8288
Phone: +358-3-3140-8200
Email: sales@vlsi.fi
URL: http://www.vlsi.fi/



14 联系信息

VLSI Solution Oy
入口G，赫尔米安卡
图8号2楼，FI-
33720 坦佩雷，芬
兰

传真：+358-3-3140-828
8 电话：+358-3-3140-8200
邮箱：sales@vlsi.fi
网址：http://www.vlsi.fi/